# Final_Project

*Ankit, Nathan, Chris*

*7/25/2020*

## Contents

```r
# Randomization
# read in the csv
zip_pop_df <- read.csv("C:/Users/Games/Desktop/Berkeley DS/Summer 2020/w241/finalProject/Produ

# puts 0s for zipcodes
zip_pop_df$zip_code <- as.character(zip_pop_df$zip_code)

zip_pop_df_2 <- data.frame(zip_code = rep(NA,length(zip_pop_df$zip_code)),
                   y2016 = rep(NA,length(zip_pop_df$zip_code)),
                   stringsAsFactors = FALSE)


for(i in 1:nrow(zip_pop_df)) {
  row <- zip_pop_df[i,]
  # do stuff with row
  if (nchar(row$zip_code) == 3){
    row$zip_code <- paste("00",row$zip_code, sep="")
    }
  else if (nchar(row$zip_code) == 4){
    row$zip_code <- paste("0",row$zip_code, sep="")
  }
  else{
    row$zip_code <- row$zip_code
  }
  zip_pop_df_2[i,] <- row
  }


summary(zip_pop_df_2)
```

```
##    zip_code             y2016
##  Length:33120       Min.   :     0
##  Class :character   1st Qu.:   718
##  Mode  :character   Median :  2808
##                     Mean   :  9724
##                     3rd Qu.: 13178
##                     Max.   :115104
```

```r
zip_pop_df_3 <- zip_pop_df_2[zip_pop_df_2$y2016>=10000,]


zip_vec <- sample(zip_pop_df_3$zip_code,size=99)

# filter dataframe for everything larger than 10,000 as mean is 9724



application <- function(subjects) {
```

```
  sample(c(rep("offerup",subjects),rep("letgo", subjects), rep("cragslist", subjects)))
}

app_col <- application(33)


treatment <- function(subjects) {

  sample(c(rep("control",subjects),rep("treatment_1", subjects), rep("treatment_2", subjects)))
}


treatment_col <- treatment(33)

products <- function(subjects) {

  sample(c(rep("Speaker",subjects),rep("Bycicle from Rocket Next", subjects), rep("Keurig", sub
           rep("Logitech Pro Gaming Mouse", subjects), rep("Kid's Study Table", subjects), rep
           rep("Love Letter Board Game", subjects), rep("Patio Chairs", subjects), rep("Apple M
           rep("Apple Keyboard", subjects)))
}

products_col <- products(9)

final_df <- data.frame(zip_vec, app_col, treatment_col, products_col)
#
# final_df["app"] <- app_col
# final_df["treatment"] <- treatment_col
# final_df["products"] <- products_col



# write_xlsx(final_df, "C:/Users/Games/Desktop/Berkeley DS/Summer 2020/w241/finalProject/zipco
```

## Setup & Pre-Experiment

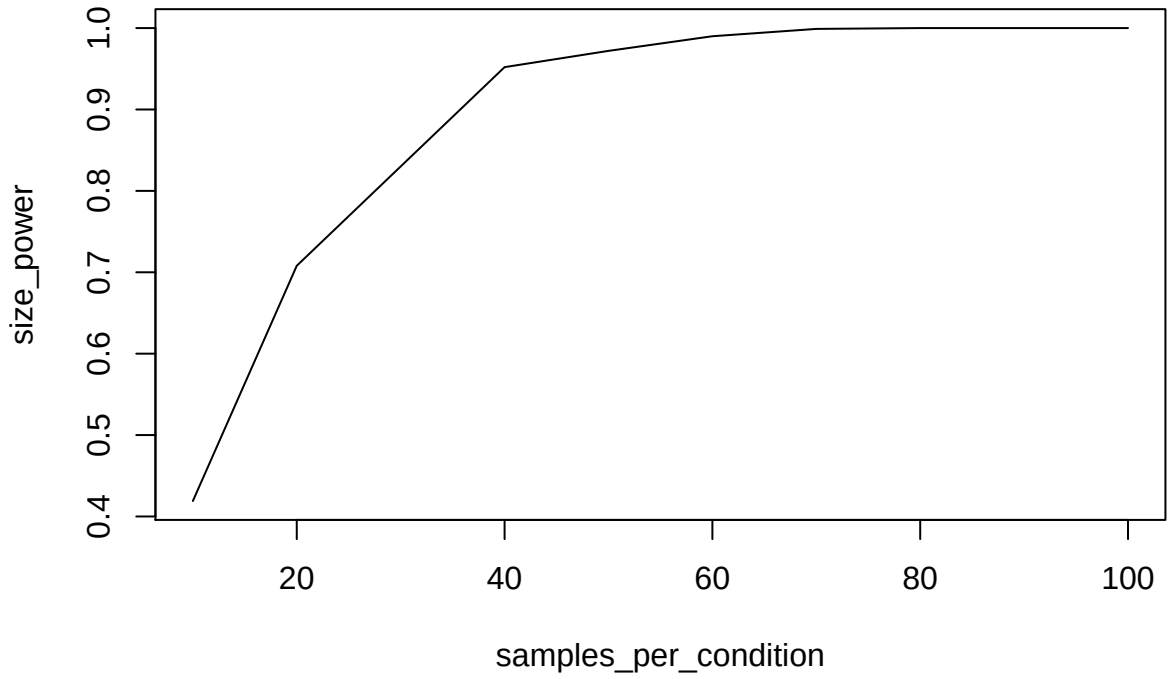
```
# Increasing sample size
samples_per_condition <- c(10, 20, 40, 50, 60, 70, 80, 90, 100)

size_power <- NA

for(i in 1:length(samples_per_condition)) {
  size_power[i] <- power_test_t(
    mean_control = 14.0, mean_treat = 18.0,
    sd_control = 5, sd_treat =  5,
    power_loops = 1000, verbose = FALSE,
```

```
    number_per_condition = samples_per_condition[i]
    )$power
}

plot(x = samples_per_condition, y = size_power, type = 'l')
```




## Results Loading

## Data Preprocessing

```
# Summary tables for information.
kable(xtable(unique(df[,c('products','Price')])))
```

| products | Price |
|---|---|
| Love Letter Board Game | 5 |
| Logitech G Pro Gaming Mouse | 10 |
| Bicycle from Rocket Next | 35 |
| Apple Keyboard | 65 |
| VTECH Sit to Stand Learning Walker | 5 |
| Kid's Study Table | 35 |
| Apple Mouse | 35 |

| products | Price |
|---|---|
| Patio Chairs | 20 |
| Retrospec Longboard | 40 |
| Speaker | 5 |
| Keurig | 50 |

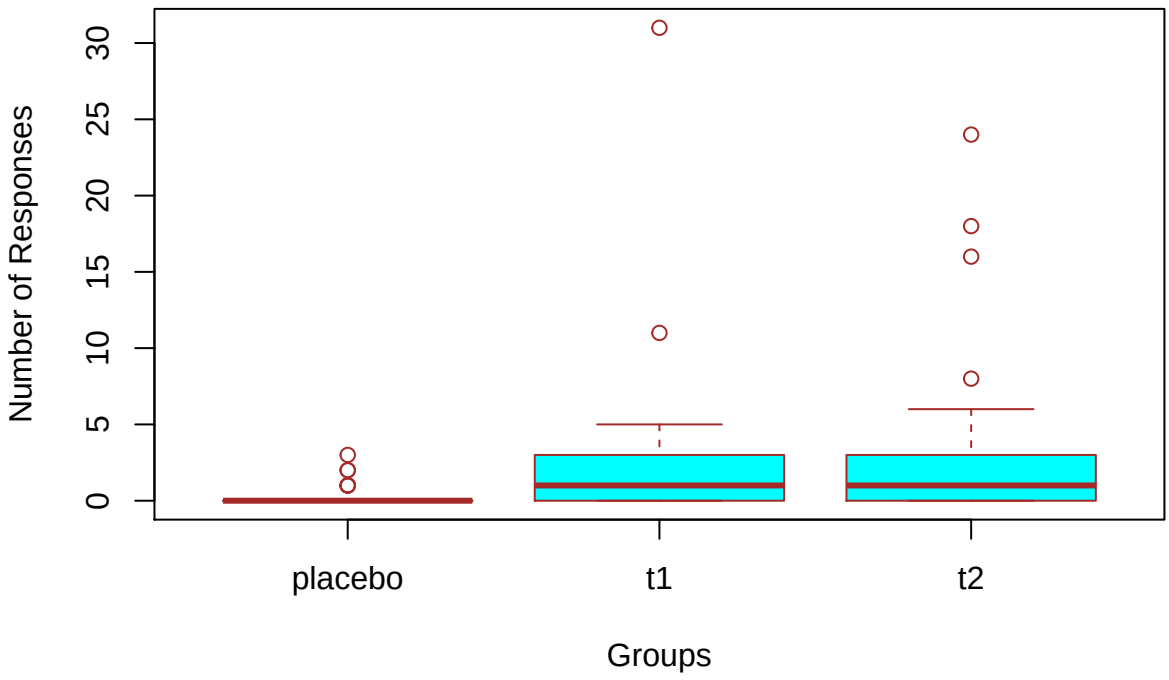
```
# Write to CSV for easy pasting into report.
#write.csv(unique(df[,c('products','Price')]),'lol.csv')
```

Table 2: Final Experimental Design

| app | treatment_final | count | total_responses | total_views |
|---|---|---|---|---|
| letgo | placebo | 10 | 5 | 176 |
| letgo | t1 | 12 | 60 | 567 |
| letgo | t2 | 11 | 80 | 695 |
| cragslist | placebo | 12 | 2 | 0 |
| cragslist | t1 | 9 | 11 | 0 |
| cragslist | t2 | 12 | 15 | 0 |
| offerup | placebo | 11 | 4 | 200 |
| offerup | t1 | 12 | 17 | 498 |
| offerup | t2 | 10 | 13 | 395 |

## EDA

```
boxplot(Responses~treatment_final,
data=df,
main="Distributions of the number of responses for all groups",
xlab="Groups",
ylab="Number of Responses",
col="cyan",
border="brown"
)
```




```
# Generate Additional Summary Data Tables
kable(df[order(app,treatment_final,products),
         .(count=.N,total_responses=sum(Responses)),
         by=list(app,treatment_final,products)],
      caption = 'Overall Results')
```

Table 3: Overall Results

| app | treatment_final | products | count | total_responses |
|---|---|---|---|---|
| letgo | placebo | Apple Keyboard | 1 | 0 |
| letgo | placebo | Bicycle from Rocket Next | 1 | 0 |
| letgo | placebo | Keurig | 1 | 1 |
| letgo | placebo | Kid's Study Table | 1 | 0 |
| letgo | placebo | Logitech G Pro Gaming Mouse | 3 | 1 |
| letgo | placebo | Love Letter Board Game | 1 | 0 |
| letgo | placebo | Patio Chairs | 1 | 3 |
| letgo | placebo | Speaker | 1 | 0 |
| letgo | t1 | Apple Mouse | 1 | 0 |
| letgo | t1 | Bicycle from Rocket Next | 1 | 2 |
| letgo | t1 | Keurig | 1 | 2 |
| letgo | t1 | Kid's Study Table | 1 | 2 |
| letgo | t1 | Logitech G Pro Gaming Mouse | 1 | 2 |
| letgo | t1 | Love Letter Board Game | 3 | 2 |

| app | treatment_final | products | count | total_responses |
|---|---|---|---|---|
| letgo | t1 | Patio Chairs | 1 | 31 |
| letgo | t1 | Retrospec Longboard | 2 | 14 |
| letgo | t1 | Speaker | 1 | 5 |
| letgo | t2 | Apple Keyboard | 1 | 0 |
| letgo | t2 | Apple Mouse | 1 | 2 |
| letgo | t2 | Bicycle from Rocket Next | 2 | 9 |
| letgo | t2 | Keurig | 1 | 1 |
| letgo | t2 | Kid's Study Table | 2 | 7 |
| letgo | t2 | Patio Chairs | 1 | 18 |
| letgo | t2 | Retrospec Longboard | 1 | 24 |
| letgo | t2 | Speaker | 1 | 16 |
| letgo | t2 | VTECH Sit to Stand Learning Walker | 1 | 3 |
| cragslist | placebo | Apple Keyboard | 2 | 0 |
| cragslist | placebo | Apple Mouse | 1 | 0 |
| cragslist | placebo | Bicycle from Rocket Next | 1 | 0 |
| cragslist | placebo | Keurig | 2 | 1 |
| cragslist | placebo | Kid's Study Table | 1 | 0 |
| cragslist | placebo | Logitech G Pro Gaming Mouse | 1 | 1 |
| cragslist | placebo | Patio Chairs | 1 | 0 |
| cragslist | placebo | Speaker | 1 | 0 |
| cragslist | placebo | VTECH Sit to Stand Learning Walker | 2 | 0 |
| cragslist | t1 | Apple Mouse | 1 | 4 |
| cragslist | t1 | Bicycle from Rocket Next | 1 | 1 |
| cragslist | t1 | Kid's Study Table | 1 | 0 |
| cragslist | t1 | Logitech G Pro Gaming Mouse | 2 | 0 |
| cragslist | t1 | Retrospec Longboard | 2 | 6 |
| cragslist | t1 | Speaker | 1 | 0 |
| cragslist | t1 | VTECH Sit to Stand Learning Walker | 1 | 0 |
| cragslist | t2 | Apple Keyboard | 2 | 1 |
| cragslist | t2 | Keurig | 2 | 4 |
| cragslist | t2 | Kid's Study Table | 1 | 1 |
| cragslist | t2 | Logitech G Pro Gaming Mouse | 1 | 2 |
| cragslist | t2 | Love Letter Board Game | 2 | 0 |
| cragslist | t2 | Patio Chairs | 3 | 7 |
| cragslist | t2 | VTECH Sit to Stand Learning Walker | 1 | 0 |
| offerup | placebo | Apple Keyboard | 2 | 0 |
| offerup | placebo | Apple Mouse | 1 | 0 |
| offerup | placebo | Logitech G Pro Gaming Mouse | 1 | 2 |
| offerup | placebo | Love Letter Board Game | 1 | 0 |
| offerup | placebo | Patio Chairs | 1 | 0 |
| offerup | placebo | Retrospec Longboard | 1 | 0 |
| offerup | placebo | Speaker | 2 | 0 |
| offerup | placebo | VTECH Sit to Stand Learning Walker | 2 | 2 |
| offerup | t1 | Apple Keyboard | 1 | 2 |
| offerup | t1 | Apple Mouse | 2 | 2 |
| offerup | t1 | Bicycle from Rocket Next | 1 | 1 |

| app | treatment_final | products | count | total_responses |
|---|---|---|---|---|
| offerup | t1 | Kid's Study Table | 2 | 2 |
| offerup | t1 | Love Letter Board Game | 1 | 0 |
| offerup | t1 | Patio Chairs | 1 | 4 |
| offerup | t1 | Retrospec Longboard | 2 | 5 |
| offerup | t1 | Speaker | 1 | 1 |
| offerup | t1 | VTECH Sit to Stand Learning Walker | 1 | 0 |
| offerup | t2 | Apple Mouse | 2 | 3 |
| offerup | t2 | Bicycle from Rocket Next | 2 | 2 |
| offerup | t2 | Keurig | 2 | 5 |
| offerup | t2 | Love Letter Board Game | 1 | 0 |
| offerup | t2 | Retrospec Longboard | 1 | 0 |
| offerup | t2 | Speaker | 1 | 3 |
| offerup | t2 | VTECH Sit to Stand Learning Walker | 1 | 0 |

```
kable(df[order(app,treatment_final),
         .(count=.N,
           total_responses=sum(Responses),
           total_views=sum(Views)),
         by=list(app,treatment_final)],
      caption = 'Overall Results')
```

Table 4: Overall Results

| app | treatment_final | count | total_responses | total_views |
|---|---|---|---|---|
| letgo | placebo | 10 | 5 | 176 |
| letgo | t1 | 12 | 60 | 567 |
| letgo | t2 | 11 | 80 | 695 |
| cragslist | placebo | 12 | 2 | 0 |
| cragslist | t1 | 9 | 11 | 0 |
| cragslist | t2 | 12 | 15 | 0 |
| offerup | placebo | 11 | 4 | 200 |
| offerup | t1 | 12 | 17 | 498 |
| offerup | t2 | 10 | 13 | 395 |

## Analysis I: Zip Codes are Enrolled Entities; Integer Outcome Variable

- All data included, including Craigslist.

```
stargazer(lm1.0,
          lm1.1,
          lm1.4,
          lm1.2,
          lm1.3,
```

```
          se = list(
          sqrt(diag(vcovHC(lm1.0))),
          sqrt(diag(vcovHC(lm1.1))),
          sqrt(diag(vcovHC(lm1.4))),
          sqrt(diag(vcovHC(lm1.2))),
          sqrt(diag(vcovHC(lm1.3)))),
          type='latex',
          omit=c('products'),
          omit.stat='f',
          add.lines=list(c('Product Fixed Effects?','No','Yes','No','No','No')),
          dep.var.caption  = 'Dependent Variable: Responses',
          title='Analysis I: Raw Response Volumes vs. Product Images + Covariates',
          font.size = "tiny")
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu % Date and time: Mon, Aug 10, 2020 - 8:44:27 PM

Table 5: Analysis I: Raw Response Volumes vs. Product Images + Covariates

| | Dependent Variable: Responses | | | | |
|---|---|---|---|---|---|
| | Responses | | | | |
| | (1) | (2) | (3) | (4) | (5) |
| treatment_finalt1 | 2.333** | 2.015 | 2.120** | 2.129** | 4.471 |
| | (0.984) | (1.269) | (0.948) | (0.949) | (2.730) |
| treatment_finalt2 | 2.939*** | 2.787*** | 2.840*** | 2.820*** | 6.822** |
| | (0.992) | (1.082) | (0.917) | (0.902) | (2.681) |
| appcragslist | | | −3.439*** | −3.500*** | −0.399 |
| | | | (1.275) | (1.287) | (0.343) |
| appofferup | | | −3.278** | −3.295** | −0.159 |
| | | | (1.290) | (1.298) | (0.388) |
| Population | | | | 0.00001 | 0.00002 |
| | | | | (0.00003) | (0.00004) |
| treatment_finalt1:appcragslist | | | | | −3.416 |
| | | | | | (2.789) |
| treatment_finalt2:appcragslist | | | | | −5.805** |
| | | | | | (2.775) |
| treatment_finalt1:appofferup | | | | | −3.344 |
| | | | | | (2.850) |
| treatment_finalt2:appofferup | | | | | −5.963** |
| | | | | | (2.796) |
| Constant | 0.333** | −0.819 | 2.676*** | 2.365* | 0.016 |
| | (0.130) | (0.565) | (0.966) | (1.294) | (0.940) |
| Product Fixed Effects? | No | Yes | No | No | No |
| Observations | 99 | 99 | 99 | 99 | 99 |
| $R^2$ | 0.074 | 0.261 | 0.190 | 0.191 | 0.249 |
| Adjusted $R^2$ | 0.055 | 0.158 | 0.156 | 0.148 | 0.173 |
| Residual Std. Error | 4.543 (df = 96) | 4.289 (df = 86) | 4.294 (df = 94) | 4.314 (df = 93) | 4.249 (df = 89) |
| *Note:* | | | | | *p<0.1; **p<0.05; ***p<0.01 |

```
# F test for two most relevant models.
anova(lm1.3,lm1.4, test='F')
```

```
## Analysis of Variance Table
##
## Model 1: Responses ~ treatment_final * (app) + Population
## Model 2: Responses ~ treatment_final + app
```

```
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     89 1607.1
## 2     94 1733.4 -5   -126.36 1.3996 0.2321
```

```
# Confidence intervals for reporting.
confint(lm1.0_results)
```

```
##                       2.5 %   97.5 %
## (Intercept)       0.07507869 0.591588
## treatment_finalt1 0.38092400 4.285743
## treatment_finalt2 0.97009679 4.908691
```

```
confint(lm1.1_results)
```

```
##                                                 2.5 %     97.5 %
## (Intercept)                                -1.9428943  0.3039747
## treatment_finalt1                          -0.5072819  4.5368055
## treatment_finalt2                           0.6366339  4.9369502
## productsApple Mouse                        -1.2707705  1.7053737
## productsBicycle from Rocket Next           -1.3524603  2.5043901
## productsKeurig                             -0.7750515  1.9809217
## productsKid's Study Table                  -1.3860190  2.0428444
## productsLogitech G Pro Gaming Mouse        -0.7842665  2.2385022
## productsLove Letter Board Game             -2.3111778  0.7457810
## productsPatio Chairs                       -1.1363172 13.4026385
## productsRetrospec Longboard                -1.8064440 10.4093292
## productsSpeaker                            -1.2677957  5.8805220
## productsVTECH Sit to Stand Learning Walker -1.5258776  1.5225972
```

```
confint(lm1.4_results)
```

```
##                       2.5 %     97.5 %
## (Intercept)        0.7573835  4.5952680
## treatment_finalt1  0.2379587  4.0021148
## treatment_finalt2  1.0191539  4.6609931
## appcragslist      -5.9702387 -0.9073348
## appofferup        -5.8396669 -0.7154802
```

```
confint(lm1.2_results)
```

```
##                           2.5 %        97.5 %
## (Intercept)       -2.038995e-01  4.934706e+00
## treatment_finalt1  2.440196e-01  4.013892e+00
## treatment_finalt2  1.028714e+00  4.610880e+00
## appcragslist      -6.056537e+00 -9.436899e-01
## appofferup        -5.873197e+00 -7.166489e-01
## Population        -5.014224e-05  7.561485e-05
```

```
confint(lm1.3_results)
```

```
##                                      2.5 %       97.5 %
## (Intercept)                  -1.851800e+00  1.8836312873
```

```
## treatment_finalt1              -9.529567e-01  9.8946671994
## treatment_finalt2               1.494860e+00 12.1496003073
## appcragslist                   -1.081002e+00  0.2833778644
## appofferup                     -9.308171e-01  0.6121969033
## Population                     -5.475988e-05  0.0000934973
## treatment_finalt1:appcragslist -8.957944e+00  2.1260652264
## treatment_finalt2:appcragslist -1.131781e+01 -0.2912952143
## treatment_finalt1:appofferup   -9.007099e+00  2.3186149737
## treatment_finalt2:appofferup   -1.151867e+01 -0.4072303932
```

### Analysis II: Views are Individual Enrollees; CACE, Binary Outcome Variable

- Craigslist data excluded (N/A)

```
# Stargazer to format.
# Latex / PDF format for screengrabs.
stargazer(lm4.0,
          lm4.1,
          lm4.2,
          lm4.3,
          se = list(
          sqrt(diag(vcovCL(lm4.0,cluster = df5_f[,zip_vec]))),
          sqrt(diag(vcovCL(lm4.1,cluster = df5_f[,zip_vec]))),
          sqrt(diag(vcovCL(lm4.2,cluster = df5_f[,zip_vec]))),
          sqrt(diag(vcovCL(lm4.3,cluster = df5_f[,zip_vec])))),
          omit=c('zip_vec','products'),
          type='latex',
          omit.stat='f',
          dep.var.caption  = 'Dependent Variable: Response for each View Observation',
          title='Analysis II: Responses for Each View vs. Product Images + Covariates',
          add.lines=list(c('Product Fixed Effects:','No','Yes','No','No'),
                         c('df:','2528','2527','2517','2515')),
          df = FALSE,
          font.size='tiny')
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu % Date and time: Mon, Aug 10, 2020 - 8:44:27 PM

Table 6: Analysis II: Responses for Each View vs. Product Images + Covariates

| | Dependent Variable: Response for each View Observation | | | |
|---|---|---|---|---|
| | outcome | | | |
| | (1) | (2) | (3) | (4) |
| treatment_finalt1 | 0.048** | 0.050** | 0.044** | 0.077** |
| | (0.024) | (0.025) | (0.021) | (0.037) |
| treatment_finalt2 | 0.061*** | 0.071*** | 0.050*** | 0.087*** |
| | (0.019) | (0.020) | (0.016) | (0.022) |
| appofferup | | −0.067*** | −0.066*** | −0.008 |
| | | (0.016) | (0.019) | (0.014) |
| treatment_finalt1:appofferup | | | | −0.063 |
| | | | | (0.039) |
| treatment_finalt2:appofferup | | | | −0.074*** |
| | | | | (0.025) |
| Constant | 0.024*** | 0.033 | 0.059*** | 0.028*** |
| | (0.007) | (0.047) | (0.014) | (0.009) |
| Product Fixed Effects: | No | Yes | No | No |
| df: | 2528 | 2527 | 2517 | 2515 |
| Observations | 2,531 | 2,531 | 2,531 | 2,531 |
| $R^2$ | 0.006 | 0.041 | 0.023 | 0.025 |
| Adjusted $R^2$ | 0.006 | 0.036 | 0.021 | 0.023 |
| Residual Std. Error | 0.256 | 0.252 | 0.254 | 0.253 |
| *Note:* | | | | *p<0.1; **p<0.05; ***p<0.01 |

```
# F test of the two most relevant models.
anova(lm4.3, lm4.2,test='F')
```

```
## Analysis of Variance Table
##
## Model 1: outcome ~ treatment_final * app
## Model 2: outcome ~ treatment_final + app
##   Res.Df    RSS Df Sum of Sq     F  Pr(>F)
## 1   2525 162.21
## 2   2527 162.60 -2  -0.38507 2.997 0.05011 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Compute confidence intervals for reporting purposes.
confint(lm4.0_results)
```

```
##                          2.5 %     97.5 %
## (Intercept)       0.0100537877 0.03781855
## treatment_finalt1 0.0003552021 0.09637340
## treatment_finalt2 0.0244469248 0.09832294
```

```
confint(lm4.1_results)
```

```
##                                                  2.5 %      97.5 %
## (Intercept)                                -0.059696849  0.12517646
## treatment_finalt1                           0.001814804  0.09795244
## treatment_finalt2                           0.031425859  0.11047540
## appofferup                                 -0.098828898 -0.03490510
## productsApple Mouse                        -0.093412999  0.10291751
## productsBicycle from Rocket Next           -0.120270360  0.08368575
## productsKeurig                             -0.111356610  0.08671031
## productsKid's Study Table                  -0.144493710  0.05519876
## productsLogitech G Pro Gaming Mouse        -0.083328805  0.12772793
## productsLove Letter Board Game             -0.112890403  0.07813016
## productsPatio Chairs                       -0.039535676  0.18334378
## productsRetrospec Longboard                -0.068201744  0.12610736
## productsSpeaker                            -0.072116828  0.12679251
## productsVTECH Sit to Stand Learning Walker -0.070035975  0.14263422
```

```
confint(lm4.2_results)
```

```
##                          2.5 %      97.5 %
## (Intercept)         0.031494525  0.08692967
## treatment_finalt1   0.002122178  0.08607660
## treatment_finalt2   0.018409188  0.08187470
## appofferup         -0.102869823 -0.02976766
```

```
confint(lm4.3_results)
```

```
##                                    2.5 %      97.5 %
## (Intercept)                  0.011263141  0.04555504
## treatment_finalt1            0.004111055  0.15071097
## treatment_finalt2            0.043345512  0.13005213
## appofferup                  -0.035064986  0.01824680
```

```
## treatment_finalt1:appofferup -0.139963935  0.01341500
## treatment_finalt2:appofferup -0.122886864 -0.02468800
```