# HW2: Rocket Fuel Analysis

Annie Chen

2023-10-30

Load the packages we want.

```
### Load all packages needed
library(dplyr)
library(knitr)
library(ggplot2)
```

Read the data.

```
### Import the rocketfuel deciles data
rocketfuel_data = read.csv("/Users/anniechen/Desktop/rocketfuel_deciles.csv", header = TRUE)
```

Summarize the data.

```
summary = summary(rocketfuel_data)
kable(summary, digits = 2)
```

| user_id | test | converted | tot_impr | mode_impr_day | mode_impr_hour | tot_impr_decile |
|---|---|---|---|---|---|---|
| Min. : 900000 | Min. :0.00 | Min. :0.00000 | Min. : 1.00 | Min. :1.000 | Min. : 0.00 | Min. : 1.000 |
| 1st Qu.:1143190 | 1st Qu.:1.00 | 1st Qu.:0.00000 | 1st Qu.: 4.00 | 1st Qu.:2.000 | 1st Qu.:11.00 | 1st Qu.: 3.000 |
| Median :1313725 | Median :1.00 | Median :0.00000 | Median : 13.00 | Median :4.000 | Median :14.00 | Median : 5.000 |
| Mean :1310692 | Mean :0.96 | Mean :0.02524 | Mean : 24.82 | Mean :4.026 | Mean :14.47 | Mean : 5.448 |
| 3rd Qu.:1484088 | 3rd Qu.:1.00 | 3rd Qu.:0.00000 | 3rd Qu.: 27.00 | 3rd Qu.:6.000 | 3rd Qu.:18.00 | 3rd Qu.: 8.000 |
| Max. :1654483 | Max. :1.00 | Max. :1.00000 | Max. :2065.00 | Max. :7.000 | Max. :23.00 | Max. :10.000 |

We can see that treatment takes on the values 0 or 1. There are approximately 590,000 users identified, about 15,000 bought the new handbag.

- The subgroups are deciles or groups formed based on different number of times the individual was targeted for ads (tot_impr).
- The pre-experiment variables include day of the week on which the user encountered the most number of impressions (mode_impr_day) and the hour of the day in which the user encountered the most number of impressions (mode_impr_hour).
- The outcome is “converted”, which indicates the impact or effect of the ad campaign on user purchasing behavior, thus serving as the outcome variable.

# 1. Check for Balance

```
# make balance table, looking at counts and means by treatment
balance_table = rocketfuel_data %>% group_by(test) %>%
  summarise(N=n(),
            mean_converted = mean(converted),
            mean_tot_impr = mean(tot_impr),
            mean_day = mean(mode_impr_day),
            mean_hour = mean(mode_impr_hour),
            sd_converted = sd(converted),
            sd_tot_impr = sd(tot_impr),
            sd_day = sd(mode_impr_day),
            sd_hour = sd(mode_impr_hour),
            prop = n()/588101
            )

kable(balance_table, digits = 4)
```

| test | N | mean_converted | mean_tot_impr | mean_day | mean_hour | sd_converted | sd_tot_impr | sd_day | sd_hour | prop |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 23524 | 0.0179 | 24.7611 | 3.9526 | 14.3049 | 0.1324 | 42.8607 | 1.9489 | 4.6562 | 0.04 |
| 1 | 564577 | 0.0255 | 24.8234 | 4.0286 | 14.4759 | 0.1578 | 43.7505 | 2.0062 | 4.8418 | 0.96 |

**Q1 - Check for balance:**

- In the table above, we can see that there are 4% users in the control group and 96% in treatment group.
- However, the number of times the individual was targeted for ads (tot_impr) variable is quite balance for both control and treatment group.
- The mean value for the 'converted' variable is slightly higher in test 1 (0.0255) compared to test 0 (0.0179), indicating a difference in the average conversion rate between the groups.
- There seems to be an imbalance in the 'converted' variable between the test groups, while other variables display a relatively balanced distribution among the groups.

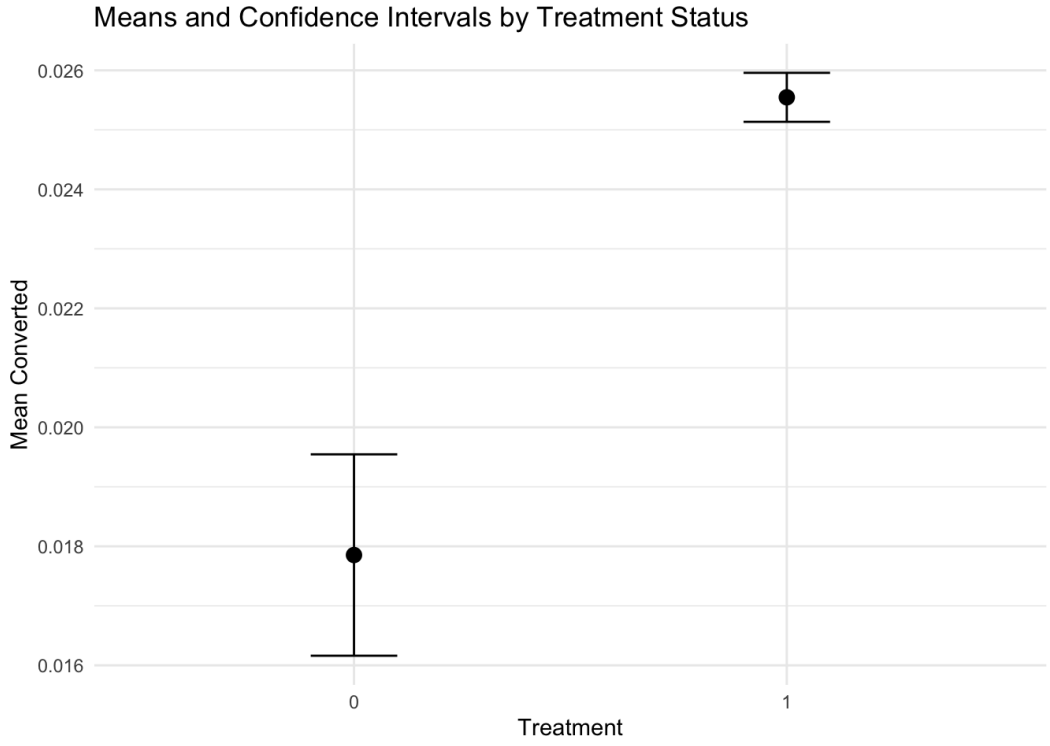
# 2. Plot Outcome

Plot the means and confidence intervals of the main outcome "converted," treatment status. In your markdown, file, interpret the result (mean and confidence intervals, noting width).

```
# Generating summary statistics by treatment status
summary_data <- rocketfuel_data %>%
  group_by(test) %>%
  summarise(
    mean_converted = mean(converted),
    sd_converted = sd(converted),
    n = n(),
    se_converted = sd_converted / sqrt(n)
  )

# Calculating 95% confidence intervals
summary_data <- summary_data %>%
  mutate(
    lower_ci = mean_converted - 1.96 * se_converted,  # Using Z-score for 95% CI
    upper_ci = mean_converted + 1.96 * se_converted   # Using Z-score for 95% CI
  )
```

```
# Plotting means and confidence intervals
ggplot(summary_data, aes(x = factor(test), y = mean_converted)) +
  geom_errorbar(aes(ymin = lower_ci, ymax = upper_ci), width = 0.2) +
  geom_point(size = 3) +
  labs(title = "Means and Confidence Intervals by Treatment Status",
       x = "Treatment", y = "Mean Converted") +
  theme_minimal()
```



**Q2 - Plot Outcome:** In the plot above, the mean value of the main outcome “converted” is 0.018 in the control group and 0.025 in the treatment group. This suggests that, on average, a higher proportion of individuals in the treatment group experienced the outcome compared to the control group.

Moreover, the confidence interval shows the range that the true population mean of “converted” is expected to lie with 95% confidence. The treatment group’s confidence interval is narrower compared to the control group, indicating that more precision in estimating the true population mean for treatment group.

# 3. Find ATE

Estimate of the Average Treatment Effect (ATE) of the ads for treatment relative to control, the associated standard error, and the 95% confidence interval on the Average Treatment Effects.

```
# make treatment a *factor* variable
rocketfuel_data = rocketfuel_data %>% mutate(test = as.factor(test))

# Running regression models to estimate ATE of different variables
  # ATE for 'converted'
  model_converted <- lm(converted ~ test, data = rocketfuel_data)

  # ATE for 'tot_impr'
  model_tot_impr <- lm(tot_impr ~ test, data = rocketfuel_data)

  # ATE for 'mode_impr_day'
  model_mode_impr_day <- lm(mode_impr_day ~ test, data = rocketfuel_data)

  # ATE for 'mode_impr_hour'
  model_mode_impr_hour <- lm(mode_impr_hour ~ test, data = rocketfuel_data)


# Extracting ATE estimates, standard errors, and confidence intervals for different variables
  # 'converted'
  ATE_estimate_converted <- coef(model_converted)["test1"]
  ATE_se_converted <- summary(model_converted)$coef["test1", "Std. Error"]
  lower_CI_converted <- ATE_estimate_converted - 1.96 * ATE_se_converted
  upper_CI_converted <- ATE_estimate_converted + 1.96 * ATE_se_converted

  # 'tot_impr'
  ATE_estimate_tot_impr <- coef(model_tot_impr)["test1"]
  ATE_se_tot_impr <- summary(model_tot_impr)$coef["test1", "Std. Error"]
  lower_CI_tot_impr <- ATE_estimate_tot_impr - 1.96 * ATE_se_tot_impr
  upper_CI_tot_impr <- ATE_estimate_tot_impr + 1.96 * ATE_se_tot_impr

  # 'mode_impr_day'
  ATE_estimate_mode_impr_day <- coef(model_mode_impr_day)["test1"]
  ATE_se_mode_impr_day <- summary(model_mode_impr_day)$coef["test1", "Std. Error"]
  lower_CI_mode_impr_day <- ATE_estimate_mode_impr_day - 1.96 * ATE_se_mode_impr_day
  upper_CI_mode_impr_day <- ATE_estimate_mode_impr_day + 1.96 * ATE_se_mode_impr_day

  # 'mode_impr_hour'
  ATE_estimate_mode_impr_hour <- coef(model_mode_impr_hour)["test1"]
  ATE_se_mode_impr_hour <- summary(model_mode_impr_hour)$coef["test1", "Std. Error"]
  lower_CI_mode_impr_hour <- ATE_estimate_mode_impr_hour - 1.96 * ATE_se_mode_impr_hour
  upper_CI_mode_impr_hour <- ATE_estimate_mode_impr_hour + 1.96 * ATE_se_mode_impr_hour
```

```
# Creating a table for ATE statistics
ATE_table <- data.frame(
  Variable = c('Converted', 'Total Impressions', 'Mode Impressions Day', 'Mode Impressions Hour'),
  ATE_estimate = c(ATE_estimate_converted, ATE_estimate_tot_impr, ATE_estimate_mode_impr_day, ATE_estimate_mod
e_impr_hour),
  SE = c(ATE_se_converted, ATE_se_tot_impr, ATE_se_mode_impr_day, ATE_se_mode_impr_hour),
  Lower_CI = c(lower_CI_converted, lower_CI_tot_impr, lower_CI_mode_impr_day, lower_CI_mode_impr_hour),
  Upper_CI = c(upper_CI_converted, upper_CI_tot_impr, upper_CI_mode_impr_day, upper_CI_mode_impr_hour)
)
kable(ATE_table, digits = 4)
```

| Variable | ATE_estimate | SE | Lower_CI | Upper_CI |
|---|---|---|---|---|
| Converted | 0.0077 | 0.0010 | 0.0056 | 0.0097 |
| Total Impressions | 0.0622 | 0.2909 | -0.5079 | 0.6324 |
| Mode Impressions Day | 0.0759 | 0.0133 | 0.0498 | 0.1021 |
| Mode Impressions Hour | 0.1710 | 0.0322 | 0.1079 | 0.2340 |

**Q3 - Find ATE:**

- The Average Treatment Effect (ATE) estimate for 'converted' is 0.0077, suggesting that, on average, the treatment ('test') is associated with a 0.77% increase in the 'converted' outcome. The standard error of this estimate is 0.0010, indicating the precision or variability in estimating the ATE.
- The 95% confidence interval for the ATE ranges from 0.0056 to 0.0097. This interval provides a range in which we can be 95% confident that the true ATE lies. The ATE for 'converted' remains statistically significant and implies a small positive effect, suggesting a potential impact of the treatment variable on the 'converted' outcome.
- For 'Total Impressions', the ATE estimate is 0.0622 with a standard error of 0.2909. The wide range of the 95% confidence interval (-0.5079 to 0.6324) suggests a larger variability in the estimate and does not provide a statistically significant impact of 'Total Impressions' on the outcome.
- Regarding 'Mode Impressions Day' and 'Mode Impressions Hour', their ATE estimates are 0.0759 and 0.1710, respectively, with standard errors of 0.0133 and 0.0322. The confidence intervals (0.0498 to 0.1021 for 'Mode Impressions Day' and 0.1079 to 0.2340 for 'Mode Impressions Hour') are relatively narrow, indicating a more precise estimation compared to 'Total Impressions'. Both variables show a statistically significant impact on the outcome, suggesting a moderate positive effect.

# 4. Subgroup Analysis

```
# make treatment a *factor* variable
rocketfuel_data = rocketfuel_data %>% mutate(test = as.factor(test))

# Grouping by deciles, test group, and summarizing statistics
summary_table <- rocketfuel_data %>%
  group_by(tot_impr_decile, test) %>%
  summarise(
            mean_converted = mean(converted),
            mean_tot_impr = mean(tot_impr),
            mean_day = mean(mode_impr_day),
            mean_hour = mean(mode_impr_hour),
            sd_converted = sd(converted),
            sd_tot_impr = sd(tot_impr),
            sd_day = sd(mode_impr_day),
            sd_hour = sd(mode_impr_hour),
    N = n(),
  ) %>%
  arrange(tot_impr_decile)
kable(summary_table, digits = 2)
```

| tot_impr_decile | test | mean_converted | mean_tot_impr | mean_day | mean_hour | sd_converted | sd_tot_impr | sd_day | sd_hour | N |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0.00 | 1.00 | 3.81 | 14.23 | 0.04 | 0.00 | 1.92 | 4.68 | 2308 |
| 1 | 1 | 0.00 | 1.00 | 4.09 | 14.42 | 0.04 | 0.00 | 1.96 | 5.13 | 54298 |
| 2 | 0 | 0.00 | 2.41 | 3.75 | 14.23 | 0.05 | 0.49 | 1.99 | 4.47 | 3249 |
| 2 | 1 | 0.00 | 2.42 | 3.92 | 14.42 | 0.05 | 0.49 | 2.01 | 4.90 | 65239 |
| 3 | 0 | 0.01 | 4.56 | 3.84 | 14.46 | 0.07 | 0.50 | 1.94 | 4.56 | 2304 |
| 3 | 1 | 0.00 | 4.56 | 3.95 | 14.55 | 0.06 | 0.50 | 1.99 | 4.79 | 50425 |
| 4 | 0 | 0.01 | 6.85 | 3.98 | 14.45 | 0.07 | 0.81 | 1.93 | 4.60 | 2490 |
| 4 | 1 | 0.00 | 6.88 | 3.96 | 14.60 | 0.06 | 0.81 | 1.99 | 4.74 | 56051 |
| 5 | 0 | 0.01 | 10.95 | 3.92 | 14.45 | 0.08 | 1.42 | 1.93 | 4.72 | 2250 |
| 5 | 1 | 0.01 | 11.05 | 3.97 | 14.54 | 0.08 | 1.41 | 2.01 | 4.77 | 60882 |

| tot_impr_decile | test | mean_converted | mean_tot_impr | mean_day | mean_hour | sd_converted | sd_tot_impr | sd_day | sd_hour | N |
|---|---|---|---|---|---|---|---|---|---|---|
| 6 | 0 | 0.01 | 15.60 | 4.10 | 14.39 | 0.09 | 1.10 | 1.95 | 4.75 | 1734 |
| 6 | 1 | 0.01 | 15.54 | 4.07 | 14.50 | 0.09 | 1.06 | 2.02 | 4.74 | 56368 |
| 7 | 0 | 0.01 | 21.23 | 4.01 | 14.25 | 0.12 | 2.19 | 1.98 | 4.60 | 2662 |
| 7 | 1 | 0.01 | 20.83 | 4.09 | 14.45 | 0.11 | 2.10 | 2.02 | 4.79 | 61873 |
| 8 | 0 | 0.02 | 28.24 | 4.02 | 14.17 | 0.14 | 2.69 | 1.94 | 4.87 | 1868 |
| 8 | 1 | 0.02 | 28.59 | 4.12 | 14.37 | 0.15 | 2.64 | 2.02 | 4.84 | 47226 |
| 9 | 0 | 0.03 | 43.52 | 4.06 | 14.26 | 0.18 | 6.77 | 1.91 | 4.73 | 2234 |
| 9 | 1 | 0.05 | 43.38 | 4.10 | 14.47 | 0.22 | 6.83 | 2.02 | 4.85 | 57194 |
| 10 | 0 | 0.08 | 118.19 | 4.16 | 14.19 | 0.28 | 80.80 | 1.94 | 4.73 | 2425 |
| 10 | 1 | 0.15 | 118.48 | 4.02 | 14.42 | 0.35 | 90.88 | 2.01 | 4.85 | 55021 |

The above summary table shows the sample size, the mean and the standard deviation of variables in the data set for both treatment and control group over the 10 deciles of total impressions. By organizing this summary table, we can easily see the treatment effects for each decile. Conversion rates increase across the deciles in both treatment and control groups. This suggests a positive correlation between exposure to ad impressions and the likelihood of a user making a purchase.

```
## Demean by subgroup means
rocketfuel_data = rocketfuel_data %>%
  group_by(test) %>%
  mutate(dm_converted = converted - mean(converted))

# Retake outcome table, with dm_converted added
outcome_dm = rocketfuel_data %>%
  group_by(tot_impr_decile,test) %>%
  summarise(cv_dm_m = mean(dm_converted),
            cv_sd_m = sd(dm_converted),
            N = n()) %>%
  mutate(cv_dm_m_se = cv_sd_m/sqrt(N))
```

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

```
# Re-calculate ATE and CI and see if the precision has improved
ate_dm = outcome_dm %>%
  mutate(cv_dm_ate = cv_dm_m - lag(cv_dm_m),
         ate_se = sqrt(cv_dm_m_se^2 + lag(cv_dm_m_se)^2),
         ate_lb = cv_dm_ate - 1.96 * ate_se,
         ate_ub = cv_dm_ate + 1.96 * ate_se)

# clean up the ate_dm table
ate_dm = ate_dm %>% filter(test != 0) %>%
  select(test, cv_dm_ate, ate_lb, ate_ub)
```

```
## Adding missing grouping variables: `tot_impr_decile`
```

```
kable(ate_dm, digits = 4)
```

| tot_impr_decile | test | cv_dm_ate | ate_lb | ate_ub |
|---|---|---|---|---|
| 1 | 1 | -0.0074 | -0.0089 | -0.0059 |
| 2 | 1 | -0.0073 | -0.0089 | -0.0056 |
| 3 | 1 | -0.0095 | -0.0125 | -0.0065 |
| 4 | 1 | -0.0091 | -0.0121 | -0.0061 |
| 5 | 1 | -0.0075 | -0.0110 | -0.0041 |
| 6 | 1 | -0.0080 | -0.0123 | -0.0037 |

| tot_impr_decile | test | cv_dm_ate | ate_lb | ate_ub |
|---|---|---|---|---|
| 7 | 1 | -0.0086 | -0.0132 | -0.0041 |
| 8 | 1 | -0.0048 | -0.0112 | 0.0017 |
| 9 | 1 | 0.0102 | 0.0025 | 0.0180 |
| 10 | 1 | 0.0542 | 0.0427 | 0.0656 |

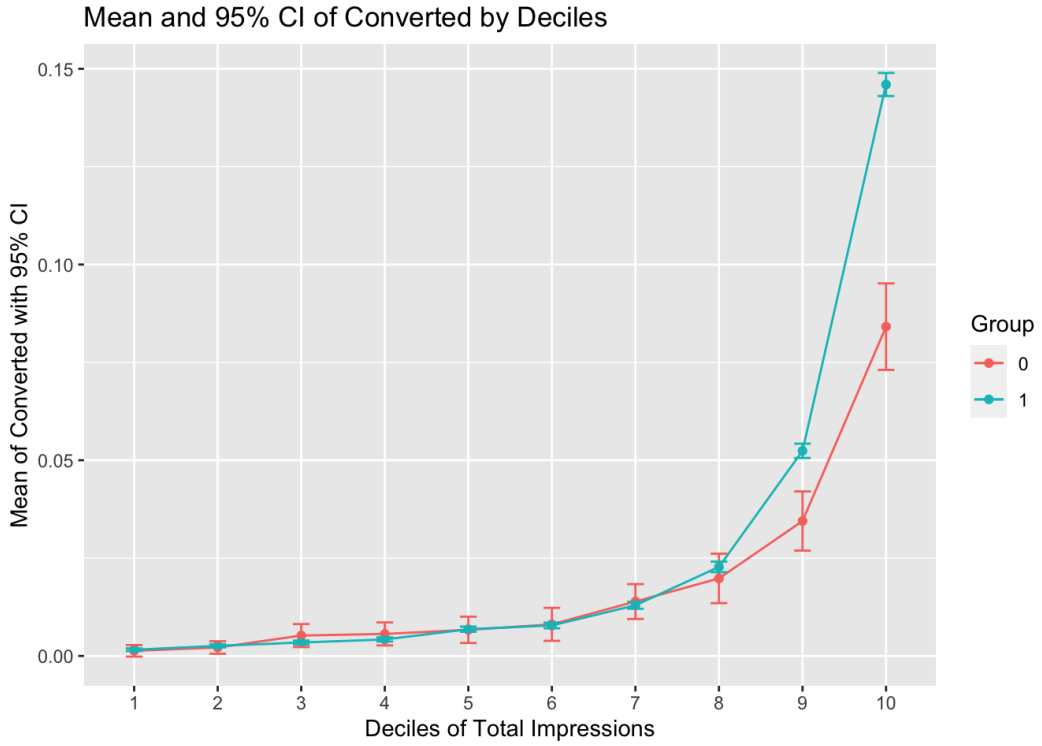
**Q4 - Subgroup Analysis:** By demeaning the subgroup means, we found that the ATE estimates for the treatment group suggest a negative correlation in the lower deciles, indicating a decrease in conversion rates compared to the control group. However, as the decile increases, the treatment group shows positive ATE estimates. Particularly, in the higher deciles, a substantial positive treatment effect is observed, signifying a notable increase in conversion rates compared to the control group.

# 5. Subgroup Plots – means + CI

Create a graph that shows the mean and 95% CI on “converted” separately for treatment and control plotted over the 10 deciles of total impressions. Briefly interpret the plot in your market down file. (Bonus: make a graph of the treatment effects and 95% CIs by decile.)

```
# Plot means and 95% CI separately for treatment and control groups
plot_data <- rocketfuel_data %>%
  group_by(tot_impr_decile, test) %>%
  summarise(mean_converted = mean(converted),
            sd_converted = sd(converted),
            N = n()) %>%
  mutate(se_converted = sd_converted / sqrt(N),
         lower_CI = mean_converted - 1.96 * se_converted,
         upper_CI = mean_converted + 1.96 * se_converted)
```

```
ggplot(plot_data, aes(x = factor(tot_impr_decile), y = mean_converted, color = factor(test))) +
  geom_line(aes(group = factor(test))) +
  geom_point(aes(group = factor(test))) +
  geom_errorbar(aes(ymin = lower_CI, ymax = upper_CI), width = 0.2) +
  labs(x = 'Deciles of Total Impressions', y = 'Mean of Converted with 95% CI', color = 'Group') +
  ggtitle('Mean and 95% CI of Converted by Deciles')
```




**Q5 - Subgroup Plots + CI:** The graph above shows the mean and the 95% CI on “converted” separately for treatment (in green) and control (in red) across the 10 deciles of total impressions. As we can see, the plot reveals a noticeable increase in the treatment effect of “converted” as the deciles progress. However, it’s also noteworthy that the 95% Confidence Interval widens with each subsequent

decile, particularly visible in the control group.

# 6. Value of advertising

Based on the overall treatment effect (question 3), what do you estimate is the increased profit per individual from running these ads? You should use information from the case to determine costs and benefits. Explain how you calculate costs and benefits.

According to the information provided in the case study:

- **Annual Revenue of Rocket Fuel:** $500 million
- **Conversion Cost:** an average of $9 for 1000 impressions.
- **Estimated Value of a Converting User:** TaskaBella estimates that a converting user is worth about $40.
- **Total Impressions Served:** 14.5 million impressions
- Incremental Revenue = Average Treatment Effect * The Value per converted user ($40)
    - 500,000,000 * 0.0077 * $40 / 1000 = $154,000
    - (This represents the total revenue gained due to the ad campaign across all individuals.)
- Estimated Costs = Cost per impression ($9/1000 impressions) * mean impressions per user in the treatment group
    - Estimated Costs = (Cost per impression) * (Total Impressions Served)
    - = ($9/1000 impressions) * 14,500,000 impressions = $130,500
    - (This represents the total cost attributed to the ad campaign.)

Calculation:

- Increased Profit Per Individual = Incremental Revenue - Estimated Costs
    - = $154,000 - $130,500 = **$23,500**
    - (This is the estimated profit gained per individual due to the ad campaign.)

It appears that the “Increased_Profit_per_Individual” shows a value of “**$23,500**”. The profit is a positive figure and implies that the ad campaign results in a gain for each individual rather than a loss.

# 7. Targeting

Now suppose you could target ads, serving only based on deciles of impression (so you can control which decile is included in the campaign, but the total impression and costs will differ by decile). Based on your heterogeneous treatment effect estimates, and with a calculation analogous to the one for (6), which decile would you want to target?

Q7 To target the ads based on deciles of impression, we could to select the decile that exhibits the most positive “cv_dm_ate” (average treatment effect by decile after demeaning of subgroup means). According to the demean subgroup analysis provided above, I would recommend to select decile “10” which has the highest positive “cv_dm_ate” value. This means that the users within decile “10” appear to exhibit the hiest positive response to the advertising campaign, making it an appealing choice for targeted ad placement.

# 8. Value of targeting

Discuss what your results imply about the value of selectively targeting ads.

Q8 By targeting at decile 10, we can re-calculate the estimated revenue per individual and the increased profit per individual for that subgroup. By doing this, we can assess the potential financial impact of targeting this specific segment.

For Decile 10 -

- ATE_decile_10 = 0.0542
- Incremental Revenue = Average Treatment Effect * The Value per converted user ($40) 500,000,000 * 0.0542 * $40 / 1000 = $1,086,400
- Estimated Costs = Cost per impression ($9/1000 impressions) * mean impressions per user in the treatment group
    - Cost per impression = $9 for 1000 impressions
    - Total Impressions Served = 14.5 million impressions
    - Estimated Costs = (Cost per impression) * (Mean Impressions Served) * (N for decile 10)

- = ($9/1000 impressions) * 118.48 impressions * 55021 = $273,292.89
- (This represents the total cost attributed to the ad campaign.)

Calculation:

- Increased Profit Per Individual = Incremental Revenue - Estimated Costs
  - Increased Profit Per Individual = Incremental Revenue - Estimated Costs
  - = $1,086,400 - $273,292.89 = **$813,107**

As we can see, by targeting the decile 10, the increased profit per individual has increased from **$23,500** (no targeting) to **$813,107** (targeting at decile 10). Therefore, we can conclude that targeting ads to specific subgroups, based on the analysis results, can significantly impact the effectiveness and efficiency of an advertising campaign with the following advantages-

- **Increased Relevance and Impact:** By identifying high-impact subgroups, selectively targeting ads can make them more relevant to those specific audiences. In this case, decile 10 respond more positively to the ads comparing to the whole group. Focusing on these subgroups can maximize the impact of the campaign on potential customers who are more likely to convert.
- **Improved Cost Efficiency:** Concentrating on specific subgroups enables more effective resource allocation, avoiding unnecessary expenses in areas that might not respond as positively to the ads.
- **Higher Conversion Rates and Returns:** Targeting the subgroups is likely to result in increased conversion rates and higher returns on investment. Ads tailored to these segments can attract and engage potential customers more effectively, resulting in improved sales or conversions.
- **Enhanced Campaign Performance:** Selective targeting allows for a more personalized and tailored advertising strategy, catering to the preferences and behavior of specific segments. This customization can significantly enhance the overall campaign performance.

In summary, the results suggest that selectively targeting ads to segments with higher treatment effects can significantly enhance the efficiency, relevance, and effectiveness of the ad campaign. This approach enables a more optimized use of resources, leading to potentially higher conversion rates and returns on investment.