Offer Engagement: Defining The Optimal Offer Type

Research Question:

Is there sufficient evidence to conclude that the take rate of BOGO offers is statically the same as discount offers?

Hypothesis:

There will be sufficient evidence at a=0.05, to say that the two offer types are statistcally the same in their redemption rates.

Context:

Answering this question benefits from data analysis for two reasons.

First, coming to a better understanding of the customer benefits the company overall. It empowers our marketing, and drives messaging.

Secondly, the goal of our membership program is to increase engagement and member value. This analysis will help measure how our loyalty program is performing towards those goals.

For this analysis in particular, understanding which offer type drives the most customer response and engagement will allow us to attaching messaging to these offers. This will help us communicate more effectively with our members.

Further more, in accomplishing the two points above, since we will selectively target the customers who are most likely to respond to a particular offer, we are minimizing the noise in our communication.

Offer type is not the only lever to minimize noise. There are other analyses needed to fine tune this process. The next steps after this is to establish customer segments and similar analysis around offer delivery.

```
library(jsonlite)
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──
```

```
## ✔ ggplot2 3.3.6     ✔ purrr   0.3.4
## ✔ tibble  3.1.7     ✔ dplyr   1.0.9
## ✔ tidyr   1.2.0     ✔ stringr 1.4.0
## ✔ readr   2.1.2     ✔ forcats 0.5.1
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter()  masks stats::filter()
## ✖ purrr::flatten() masks jsonlite::flatten()
## ✖ dplyr::lag()     masks stats::lag()
```

```
#library(qdapTools)
library(magrittr)
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:purrr':
##
##     set_names
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
library(DataExplorer)
```

Data:

I will use promotions and transaction data to evaluate the proportions of the two offer types.

The data used for this analysis was provided by Starbucks with the intent of public analysis. According to the Community Data License Agreement – Permissive – Version 1.0 (*Community Data License Agreement - Permissive - Version 1.0*, n.d.), I can use this data for my analysis.

I will be using Starbucks data that is made up of the data described above. I have downloaded the files from Ihor Muliar's (2020), profile at Kaggle.com and loaded them into R with the code block below.

Offers Data

The offers data consists of six variables and ten observations.

The variables are made up of two columns of int data (reward and difficulty), one number column (duration), two character columns (offer_type, id) and column of lists (channels).

```
str(offers)
```

```
## 'data.frame':    10 obs. of  6 variables:
##  $ reward    : int  10 10 0 5 5 3 2 0 5 2
##  $ channels  :List of 10
##   ..$ : chr  "email" "mobile" "social"
##   ..$ : chr  "web" "email" "mobile" "social"
##   ..$ : chr  "web" "email" "mobile"
##   ..$ : chr  "web" "email" "mobile"
##   ..$ : chr  "web" "email"
##   ..$ : chr  "web" "email" "mobile" "social"
##   ..$ : chr  "web" "email" "mobile" "social"
##   ..$ : chr  "email" "mobile" "social"
##   ..$ : chr  "web" "email" "mobile" "social"
##   ..$ : chr  "web" "email" "mobile"
##  $ difficulty: int  10 10 0 5 20 7 10 0 5 10
##  $ duration  : num  7 5 4 7 10 7 10 3 5 7
##  $ offer_type: chr  "bogo" "bogo" "informational" "bogo" ...
##  $ id        : chr  "ae264e3637204a6fb9bb56bc8210ddfd" "4d5c57ea9a6940dd891ad53e9dbe8da0" "3f207df678b143eea3c
ee63160fa8bed" "9b98b8c7a33c4b65b9aebfe6a799e6d9" ...
```

Transaction Data

The transaction data is comprised of 306,534 observations across seven variables.

There are two int columns (time, value.reward), four character columns (person, event, value.offer_id, and value.offer id) and one number column (value.amount).

```
str(trans)
```

```
## 'data.frame':    306534 obs. of  7 variables:
##  $ person        : chr  "78afa995795e4d85b5d9ceeca43f5fef" "a03223e636434f42ac4c3df47e8bac43" "e2127556f4f6459
2b11af22de27a7932" "8ec6ce2a7e7949b1bf142def7d0e0586" ...
##  $ event         : chr  "offer received" "offer received" "offer received" "offer received" ...
##  $ time          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ value.offer id: chr  "9b98b8c7a33c4b65b9aebfe6a799e6d9" "0b1e1539f2cc45b7b9fa7c272da2e1d7" "2906b810c7d4411
798c6938adc9daaa5" "fafdcd668e3743c1bb461111dcafc2a4" ...
##  $ value.amount  : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ value.offer_id: chr  NA NA NA NA ...
##  $ value.reward  : int  NA NA NA NA NA NA NA NA NA NA ...
```

Data Prep

Because of the format of the transactions data, the data frame needs to be cleaned up. The first step is changing the events by replacing spaces in the events with underscores.

The way the data imports into R, the offer id gets split into two different columns, depending on the event type. Ids for the events "offer_received" and "offer_viewed" get placed into the column "value.offer id". The ID for redemption events ("offer_redeemed") get placed into the column "value.offer_id". I will combine these two columns into "value.offer id" using the coalesce function. Then, I will change the name of the column with the rename function. After this process is complete I will drop the "value.offer_id" column because it is no longer needed.

```
transactions <- trans%>%
  mutate(event = str_replace(event," ","_"),
    `value.offer id` = coalesce(`value.offer id`, value.offer_id)) %>%
 rename(offer_id = `value.offer id`)%>%
 select(-value.offer_id)
```

Events and Offer Types

Below is a summary of the events and offer types by count. I will use the output to filter out any data that will not be used for this analysis.

Now that all of the offer id's are in one column, I can join the offer data frame to the transactions data frame using "offer_id" from transactions and "id" from offers. This is done in the code block below.

The join creates a data frame of 306,534 observations across 16 variables. For this analysis, I am only concerned with the person, event and offer type.

In the code below, I will drop the unneeded columns and join the offers with the transactions.

For this analysis, I can drop everything but person, event and offer type.

```
transactions <- transactions %>%
  full_join(offers, by=c("offer_id" = "id")) %>%
  select(person, event, offer_type)
head(transactions)
```

	person <chr>	event <chr>	offer_type <chr>
1	78afa995795e4d85b5d9ceeca43f5fef	offer_received	bogo
2	a03223e636434f42ac4c3df47e8bac43	offer_received	discount
3	e2127556f4f64592b11af22de27a7932	offer_received	discount
4	8ec6ce2a7e7949b1bf142def7d0e0586	offer_received	discount
5	68617ca6246f4fbc85e91a2a49552598	offer_received	bogo
6	389bc3fa690240e798340f5a15918d5c	offer_received	bogo

6 rows

Event Types

There are four types of events in the transactions data. Transactions, Offers received, offers viewed and offers completed. This analysis focuses on redemption rates of offers, so we are only concerned with offers received and offers completed.

```
transactions %>%
  group_by(event) %>%
  summarise(count = n())
```

event <chr>	count <int>
offer_completed	33579
offer_received	76277
offer_viewed	57725
transaction	138953

4 rows

Offer Types

For the purpose of this analysis we are analyzing two offer types, but our data set actually has three offer types.

```
transactions %>%
  group_by(offer_type) %>%
  summarise(count=n())
```

offer_type <chr>	count <int>
bogo	71617
discount	69898
informational	26066
NA	138953

4 rows

We can see that transaction events do not have an offer type. This is because there is a separate event type for completed offers. This means that I can filter out the transaction events.

Also, there are three offer types present in the data set. These are BOGO, discount and informational. Since this analysis will not be looking at informational, I will be filtering out that data too. I will accomplish this in the code below.

```
transactions <- transactions %>%
  filter(event != 'transactions', offer_type != 'informational') %>%
    group_by(event, offer_type) %>%
  summarise(count=n())
```

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

```
head(transactions)
```

event <chr>	offer_type <chr>	count <int>
offer_completed	bogo	15669
offer_completed	discount	17910
offer_received	bogo	30499
offer_received	discount	30543
offer_viewed	bogo	25449
offer_viewed	discount	21445

6 rows

Table Transformation

To calculate the proportions I will need to pivot the transaction data frame to a wide format. I will put the events on the columns and the offer type on the rows. This will tell me how many BOGO and discount offers were sent, received and used.

```
transactions_wide <- transactions %>%
pivot_wider( names_from = event, values_from = count) #%>%
  #mutate(redemption_rate = offer_completed/offer_received) #, offer_views = offer_viewed/offer_received)
head(transactions_wide)
```

offer_type <chr>	offer_completed <int>	offer_received <int>	offer_viewed <int>
bogo	15669	30499	25449
discount	17910	30543	21445

2 rows

Two Sample Proportions Test

This is where we evaluate the proportions of the used offer types.

Since we are evaluating if the offers are the same statistically and not if one offer type performs worse or better, I will use a two-tail hypothesis. According to *Stat 415 | Introduction to Mathmatical Statistics*. (2022), a two-tailed proportions test would use the follwing null hypothesis:

H0 : pA=pB

The alternative hypothesis would be:

Ha : pA != pB (different)

The code to compare the two offer types is below.

x is the number of completed offers, n is the number of offers received. Alternative ="two.sided" was declared since we hypothesized that the two offer types were the same.

One advantage of this test, is it will reveal if the offer types are the same or different.

One disadvantage of this technique is it does not tell us why they are different. It will take additional analysis of the other variables to uncover those details.

```
 test<-prop.test(x = transactions_wide$offer_completed, n=transactions_wide$offer_received, alternative = "two.si
ded")
 test
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  transactions_wide$offer_completed out of transactions_wide$offer_received
## X-squared = 324.99, df = 1, p-value < 2.2e-16
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.08053666 -0.06472705
## sample estimates:
##    prop 1    prop 2
## 0.5137545 0.5863864
```

The Results

The test returned a p-value less than 2.2e-16. (0.00000000000000022). Since my hypothesis stated that the two offer types would be equal at a 0.05 level, and the p-value is far below this, we will reject the null hypothesis and conclude that the two types are significantly different.

The proportion of the two groups are:

- 51% responded to BOGO offers
- 59% responded to discount offers

Discounts out perform BOGO offers by ~8% points.

Knowing this, we can say that customers purchase at a higher rate with discounts, over the customers purchasing with BOGO offers. Based on this analysis, we do not know which offer drives higher sales though. Transaction volume is important, but sales are equally as important.

Next Steps

To full understand the difference between the two offers I suggest the following analysis be done:

- I suggest evaluating the average sales for statistical significance to understand true performance.
- As stated in the context section in the beginning of this analysis, I would also evaluate the performance of channel delivery method (web, email, mobile, social delivery methods) to fine tune understanding.

References

Community Data License Agreement—Permissive—Version 1.0. (n.d.). The Linux Foundation Projects. Retrieved June 18, 2022, from https://cdla.dev/permissive-1-0/

Ihor Muliar. (2020). *Starbucks Customer Data*. https://www.kaggle.com/datasets/blacktile/starbucks-app-customer-reward-program-data

Stat 415 | Introduction to Mathmatical Statistics. (2022). https://online.stat.psu.edu/stat415/lesson/9/9.4