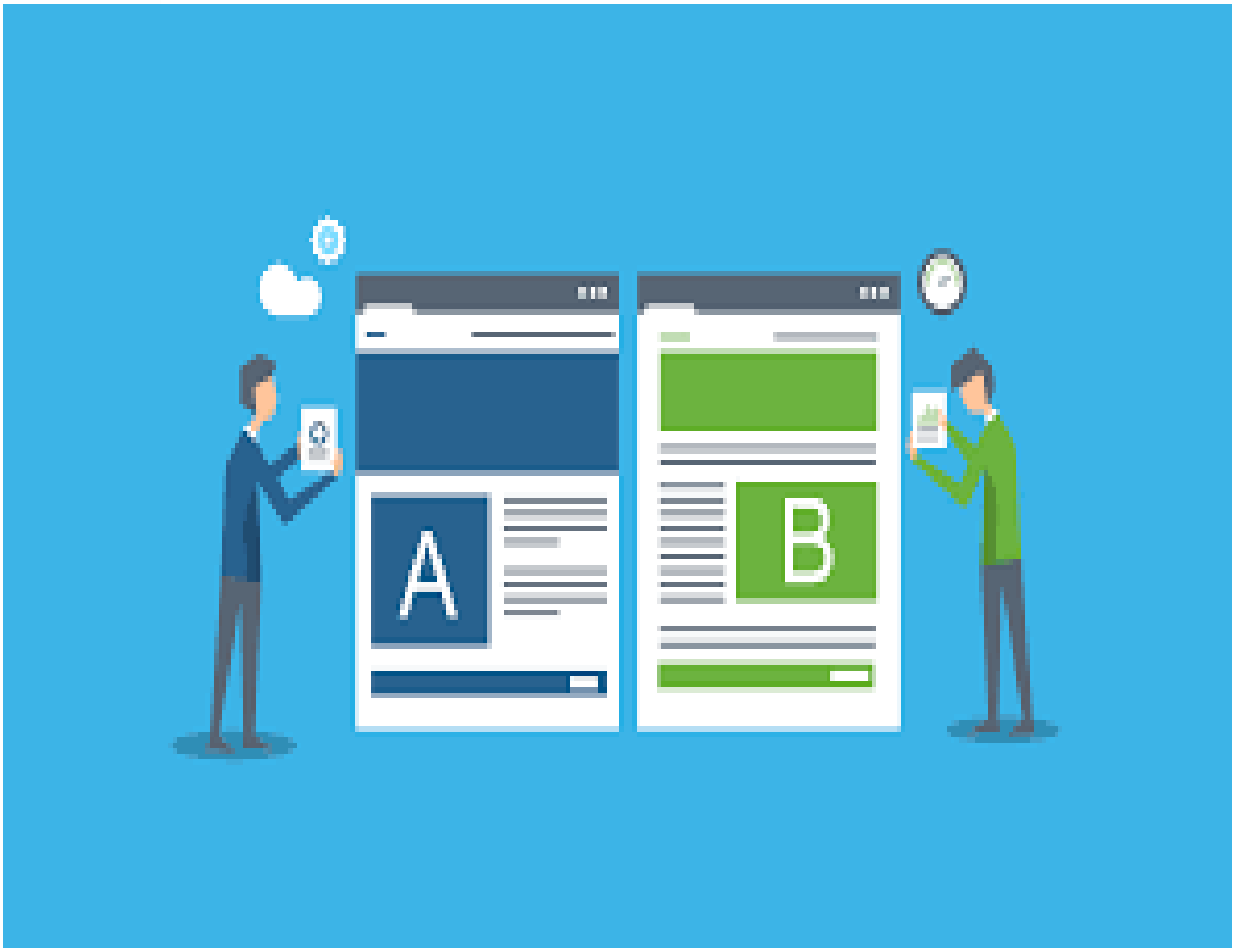


A/B TESTING NEW BANNER FOR GLOBOX

By: Sindi Albornoz Ingraham

May 12, 2023

SUMMARY

The A/B test was conducted to determine the effectiveness of a new feature on Globox's platform, using two metrics: conversion rate and average amount spent per user. The results showed that there was a statistically significant difference in the conversion rate between the control and treatment groups, with the treatment group showing a higher conversion rate. However, there was no statistically significant difference in the average amount spent per user between the two groups. Based on these findings, we recommend that Globox should not launch the new feature to all users even though it has the potential to increase conversion rates since further analysis to understand the impact of the new feature on revenue is recommended. Based on this additional analysis, stakeholders can make an informed decision about whether to launch the experiences to all users.

CONTEXT

Project Background

Globox is an online marketplace that specializes in sourcing unique and high-quality products from around the world. Their logo **" We believe that shopping should be an adventure, and we want to bring the world to your doorstep. From exotic spices and rare teas to handmade jewelry and textiles, we have a curated selection of products that you won't find anywhere else".**

GloBox is primarily known amongst its customer base for boutique fashion items and high-end decor products. However, their food and drink offerings have grown tremendously in the last few months, and the company wants to bring awareness to this product category to increase revenue.

The Growth team decides to run an A/B test that highlights key products in the food and drink category as a banner at the top of the website. The control group does not see the banner, and the test group sees it.

The setup of the A/B test is as follows:

- The experiment is only being run on the mobile website.
- A user visits the GloBox main page and is randomly assigned to either the control or test group. This is the join date for the user.
- The page loads the banner if the user is assigned to the test group and does not load the banner if the user is assigned to the control group.
- The user subsequently may or may not purchase products from the website. It could be on the same day they join the experiment, or days later. If they do make one or more purchases, this is considered a “conversion”.

Task

Analyze the results of the A/B test and provide a recommendation to the stakeholders about whether Globox should launch the experience to all users.

Metric

- Revenue
- Conversion Rate = #converted / #converted + # notconverted

Power Analysis

In this instance, power analysis was already conducted, and the test data was readily available and extracted from the database. Nonetheless, I have established the confidence level and the significance level for the study.

- **Confidence Level**: 95%
- **Significance Level**: 5% or 0.05

Minimum Detectable Effect

The targeted objective is to discern the magnitude of change between the new and old versions, as a basis for deciding whether to launch the new feature. Although the stakeholders did not provide a specific percentage, we have set a benchmark of a 2% change in conversion rate and revenue for this analysis, which will be assessed based on the average amount spent per user.

The Dataset

Globox stores its data in a relational database, which was extracted for this test using SQL queries.

To consider:

- All users should be assigned to one A/B test group.
- Not all users make a purchase.

users *user demographic information*
id *the user ID*
country *ISO 3166 alpha-3 country code*
gender *the user's gender (M = male, F = female, O = other)*

groups *user A/B test group assignment*
uid *the user ID*
group *the user's test group*
join_dt *the date the user joined the test (visited the page)*
device *the device the user visited the page on (I = iOS, A = android)*

activity *user purchase activity, containing 1 row per day that a user made a purchase*
uid *the user ID*
dt *date of purchase activity*
device *the device type the user purchased on (I = iOS, A = android)*
spent *the purchase amount in USD*

Stakeholders

- Growth Product & Engineering Team: This is the team that you work with at GloBox. The team is made up of a product manager, a user experience designer, an engineering manager, and several software engineers, and you, the data analyst. The team develops features for the GloBox website that drive growth in users and revenue.
- Leila Al-Farsi, Product Manager, Growth: Leila is the product manager for the Growth product and engineering team. Alongside Alejandro, she leads the Growth team by deciding their goals and projects, measuring their success against defined KPIs, and communicating results to other company leaders like Mei.
- Alejandro Gonzalez, User Experience Designer, Growth: Alejandro is the designer for the Growth product and engineering team. He conducts user research and designs the experience that the A/B test is evaluating.
- Mei Kim, Head of Marketing: Mei oversees the Marketing team, which works on targeting audiences with effective marketing campaigns to drive customers to the GloBox website. She collaborates frequently with Leila and Alejandro to design website experiences that will align well with the current marketing efforts.

Together, Leila, Alejandro, and Mei will decide whether or not to launch the experiment based on the results.

RESULTS

The A/B test results were analyzed primarily using Jupiter Notebook. Prior to the analysis, the dataset was extracted from a database using the following code:

```
SELECT g.uid,
      "group",
      join_dt AS join_date,
      dt AS date_of_purchase,
      COALESCE(spent, 0) AS amount_spent
FROM groups AS g
LEFT JOIN activity AS a
   USING(uid)
```

We selected the relevant columns for the analysis, including the "group" column to differentiate between the control group labeled "A" and the treatment group labeled "B". As a result of joining the different tables, some rows contained null values in the "spent" column. To ensure accurate and consistent analysis, we replaced these null values with zeros.

We imported the required packages for data manipulation, visualization, and statistical analysis in the Jupyter Notebook. Specifically, we utilized Pandas and Numpy for data handling, Matplotlib and Seaborn for visualization, and ttest_ind, stats and math modules for conducting statistical analysis. These packages enabled us to calculate the confidence intervals for proportions and means, as well as the differences between them, allowing for a more thorough analysis of the A/B test results.

```
# Importing the necessary packages

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sb
from scipy.stats import ttest_ind
import scipy.stats as stats
from math import ceil
```

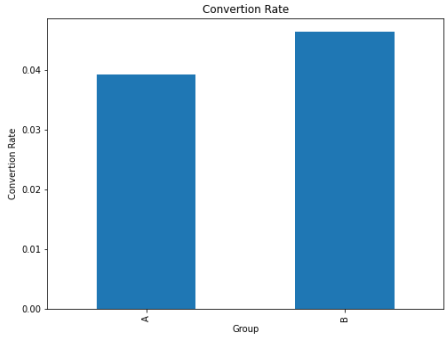
Cleaning Process

The dataset was read into a Pandas dataframe and thoroughly checked for any duplicates, data inconsistencies, or other anomalies. As the conversion rate was the initial focus of the statistical analysis, approximately 139 duplicates were identified in the dataset. After careful consideration, it was determined that their removal would not significantly impact the results, and therefore they were deleted after creating a copy of the original dataset. Furthermore, the duplicates were removed to provide a more accurate representation of the number of unique users who had converted, as some appeared more than once due to multiple purchases. The dataset passed from 49,082 to 48,943.

```
# Searching for duplicates

clean_dat = ab.copy()

data_count = clean_dat['uid'].value_counts(ascending = False)

duplicate = data_count[data_count > 1].count()

print(f'There are {duplicate} users that appear multiple times in the dataset')


# Deleteing duplicates

clean_ab = df = clean_dat.drop(clean_dat[clean_dat.duplicated(subset='uid')].index)

print(clean_ab.info())
```

Conversion Rate Confidence Interval

To gain insight into the conversion rate, we conducted initial statistical analysis by calculating the mean, standard deviation, and standard error. This provided us with a better understanding of the dataset as well as each individual group.



```
# Calculating convertion rate.

conversion_rates = clean_ab.groupby('group')['amount_spent'].apply(lambda x: (x > 0).mean())print(conversion_rates)


# Calculating standard deviation.

std_deviation = clean_ab.groupby('group')['amount_spent'].std()

print(std_deviation)


# Calculating standard of error.

std_error = clean_ab.groupby('group')['amount_spent'].sem()

print(std_error)
```

From the graph above, we can observe that the treatment group or “B” performs slightly better with approximately 4.63% conversion rate than control group or “A” with 3.92% conversion rate. However, we cannot draw conclusions yet without identifying if the results or difference is statistically significant.

In order to test for a statistically significant difference between control and treatment group, we used a two-sided z-test for proportion, since is ideal to determinate a statistical significance positive or negative results, or if there is not statistically significant between the proportions.

Hypothesis

We established hypotheses for the conversion rate and differences of proportions between the treatment and control groups. The null hypothesis assumes that there is no difference in conversion rates between the two groups, while the alternative hypothesis assumes there is a difference. Similarly, the hypothesis for the differences of proportions assumes that the conversion rate for the treatment group minus the conversion rate for the control group is equal to zero, while the alternative hypothesis assumes that it is not equal to zero.

Ho : $CR_{con} = CR_{exp}$

Ha : $CR_{con} \neq CR_{exp}$

Ho: $CR_{exp} - CR_{con} = 0$

Ha : CRexp- CRcon ≠ 0

After establishing the hypotheses, we proceeded to calculate the confidence intervals for both the control and treatment groups. The conversion rate for the control group was determined to be 3.92%, with confidence intervals of (0.0368, 0.0417). The fact that the conversion rate of the control group falls within this range indicates that the sample is representative of the population. Similarly, the conversion rate for the treatment group was calculated to be 4.63%, with confidence intervals of (0.0437, 0.0489). This falls within the range as well, indicating that the sample is also representative of the true population.

```
# Importing the necessary modules to work with the normal distribution.

import math


# assigning each value to a variable to make the calculations easier.

x_control = 955 # number of users that made a purchase

N_control = 24343 # sample size for control group

p_hat_control = x_control / N_control # This is the proportion or conversion rate for the control group

z_score_A = 1.96


#Standard error

SE_A = math.sqrt(p_hat_control * (1- p_hat_control) / N_control)


#Confidence level for control group


ci_A = (p_hat_control - z_score_A * SE_A, p_hat_control + z_score_A *  SE_A)


print(f"Convertion rate control group:", np.round(p_hat_control,4))

print("Control group confidence Interval: ({:.4f}, {:.4f})".format(ci_A[0], ci_A[1]))
```

```
x_treat = 1139 # number of users that made a purchase in the treatment group

N_treat = 24600 # sample size fro treatment group

p_hat_treatment = x_treat / N_treat  # This is the proportion or convertion rate for the treatment group

z_score_B = 1.96


#Standard error

SE_B = math.sqrt(p_hat_treatment * (1- p_hat_treatment) / N_treat)


#Confidence level for control group

ci_B = (p_hat_treatment - z_score_B * SE_B, p_hat_treatment + z_score_B *  SE_B)


print(f"Convertion rate treatment group:", np.round(p_hat_treatment,4))

print("Treatment group confidence Interval: ({:.4f}, {:.4f})".format(ci_B[0], ci_B[1]))
```

The next step of our analysis was to determine if there was statistical significance between the proportions of the control and treatment groups. To accomplish this, we calculated the confidence interval for the difference of proportions and the corresponding p-value. Our analysis found that the p-value of 0.001 was less than our predetermined significance level of 0.005, leading us to reject the null hypothesis (which assumes no difference in proportions between the two groups). This indicates that there is a statistically significant difference in the proportions of the two samples. Furthermore, the confidence interval for the difference in proportions of (0.0035, 0.0107) did not include zero, which further supports our conclusion that there is a significant difference between the groups.

```
# Importing the neccesary modules to work with the normal distribution.

from scipy.stats import norm

#Significance level

alpha = 0.05

# Now, I am going to calculate the proportion of the difference.

p_pooled = (x_control + x_treat) / (N_control + N_treat)

pooled_variance = p_pooled * (1-p_pooled) * (1/N_control + 1/N_treat) # Variance of the difference of the two groups

#Standard of error

SE = np.sqrt(pooled_variance)

#Test statistists

test_stat = (p_hat_treatment - p_hat_control)/ SE

# Critical value or z-score using the normal distribution

z_score = norm.ppf(1-alpha /2)

# Calculating the margin of error

ME = SE * z_score

#Calculating p-value

p_value = norm.sf(test_stat)*2 # We multiply here for 2 because we are using two-sided test with normal distribution.

# Calculating confidence intervals

CI = [(p_hat_treatment - p_hat_control) - SE * z_score, (p_hat_treatment - p_hat_control) + SE * z_score]

if np.abs(test_stat) >= z_score:

    print("reject the null")

    print(np.round(p_value,4))

    print("Test statistics stat: ", np.round(test_stat,4))

print("Z-Critical score: ", np.round(z_score,2))

print("P_value; ", np.round(p_value,4))

print("Confidence Interval of 2 sample for proportion: " , np.round(CI,4))
```

After analyzing the conversion rate results and rejecting the null hypothesis, we proceeded to examine the revenue, or the average amount spent per user to determine the statistical significance of launching the new feature as part of our A/B testing objectives.

Revenue confidence Interval

We proceeded to perform a hypothesis test and confidence interval analysis for the average revenue generated per user. We utilized the original dataset with duplicates since the average required all the purchases the users made during the test.

Upon calculating the average revenue generated per user, we observed that the treatment group had a slightly higher average revenue of 3.391 compared to the control group's average revenue of 3.375. However, without conducting a hypothesis test and confidence interval, we could not draw any conclusions from these observations.

Hypothesis

- Ho : $\mu con = \mu exp$
- Ha : $\mu con \neq \mu exp$
- Ho: $\mu exp - \mu con = 0$
- Ha : $\mu exp - \mu con \neq 0$

After establishing the hypotheses for the average amount spent per user, we calculated the confidence intervals for each group. The sample mean for the control group was slightly lower than the treatment group at 3.37. The confidence interval of (3.048, 3.685) gave us reasonable confidence that the true population mean fell within that range. However, we noted that our sample mean did not fall within our confidence interval, leading us to suspect that there wasn't a statistically significant difference in this group.

For the treatment group, we were 95% confident that the true population mean of the average amount spent per user fell within the interval (3.071, 3.689). Here, our sample mean of 3.380 fell within the range. To determine the statistical significance of the mean difference, we calculated the confidence interval of the mean difference and its corresponding p-value.

Code for confidence interval of the control group.

```
# Importing the t stat since we are going to use t distribution for mean
from scipy.stats import t

#Subsetting the data to incluide only the control group
control = ab[ab['group']=='A']

#Calculating some statistics over control group

NC= control.shape[0]
sample_mean = control['amount_spent'].mean()
sample_std = control['amount_spent'].std()

#Calculating the standard error of the mean
SOE = sample_std / np.sqrt(NC)

#Claculating the t-critical value, in this case 95% confidence level
t_critical = t.ppf(0.975, NC-1)

#Margin of error
MOE = t_critical * SOE

#Confidence interval
CIC = (sample_mean - MOE, sample_mean + MOE)
print(f"Sample mean; {sample_mean:.2f}")
print(f"Sample standard deviation : {sample_std:.2f}")
print(f"Standard of error: {SOE:.2f}")
print(f"t-critical value: {t_critical:.2f}")
print(f"Margin of error: {MOE:.2f}")
print(f"95% Confidence Interval: {np.round(CIC, 3)}")
```

Confidence interval for treatment group

```
#Subsetting the data to incluide only the control group

treatment = ab[ab['group']=='B']


#Calculating some statistics over control group


NT= treatment.shape[0]

sample_mean_t = treatment['amount_spent'].mean()

sample_std_t = treatment['amount_spent'].std()


#Calculating the standard error of the mean

SET = sample_std_t / np.sqrt(NT)


#Claculating the t-critical value, in this case 95% confidence level

t_critical_t = t.ppf(0.975, NT-1)


#Margin of error

MET = t_critical * SET


#Confidence interval


CIT = (sample_mean_t - MET, sample_mean_t + MET)


print(f"Sample mean: {sample_mean_t:.3f}")

print(f"Sample standard deviation : {sample_std_t:.2f}")

print(f"Standard of error: {SET:.2f}")

print(f"t-critical value: {t_critical_t:.2f}")

print(f"Margin of error: {MET:.2f}")

print(f"95% Confidence Interval: {np.round(CIT, 3)}")
```

After completing the analysis of the difference of mean, the next step was to calculate the confidence interval and p-value to determine the statistical significance of the difference. Our results showed that the p-value (0.952) was greater than the predetermined significance level (0.05), indicating that there was not enough evidence to reject the null hypothesis. In other words, we failed to reject the hypothesis that the difference between the mean of the two groups was equal to the true population mean. This suggests that any observed difference between the means may be due to chance or random variation, rather than a significant difference between the two groups.

It is important to note that a failure to reject the null hypothesis does not necessarily indicate that the null hypothesis is true. It simply means that there is insufficient evidence to support the alternative hypothesis.

```
# Calculating the confidence interval for the difference of mean using unequal variance and t-distribution


diff = sample_mean_t - sample_mean


#standard error of the difference

se_diff = np.sqrt((sample_std**2/len(control)) + (sample_std_t**2/len(treatment)))


#t-statistics and p-value

t_stat, p_value = ttest_ind(treatment['amount_spent'], control['amount_spent'], equal_var =False)


#Confidence interal for the difference

ci_low,ci_high = diff - 1.96 * se_diff,diff + 1.96 * se_diff


print(f"Difference in mean: {diff:.3f}")

print(F"95% Confidence Interval: {np.round(ci_low,3), np.round(ci_high, 3)}")

print(f"p-value:  {p_value: .3f}")
```

CONCLUSION AND RECOMMENDATION

- Based on the results of the A/B test, we can conclude that the new feature had a significant impact on the conversion rate, with the treatment group 4.63% conversion rate outperforming the control group 3.92%. However, when considering the average amount spent per user, there was no significant difference between the two groups.

- Given the targeted objective of a 2% change in conversion rate and revenue, it seems that the new feature has met this benchmark for conversion rate. However, the impact on revenue is unclear given the lack of difference in the average amount spent per user.

Recommendation

We recommend further analysis to understand the impact of the new feature on revenue, such as examining customer behavior and engagement with the new feature. Based on this additional analysis, stakeholders can make an informed decision about whether to launch the experience to all users.