Pacmann

SEKOLAH DATA PACMANN

Website of the Grocery Store Chain

Experimental Design and A/B Testing

-Siti Uyun Mubarak-

I. Introduction and Background

There is a large grocery chain. The company's goal is to drive more customers to download the mobile app and register for the loyalty program. The manager is curious if changing the link to a button of the app store will improve the user's ability to download the app. Here is the existing link button of the app store.

The manager asked to create an A/B testing plan for changing the link to a button of the app store with the expectation it will the user's interest to download the app.

II. Setting Up Problem

1. Experiment goal

To see if changes to the link to a button of the app store can increase user interest in downloading the mobile application.

2. Choosing Metrics

Goal Metrics : User counts

User count means the number of people who have downloaded the app and have installed it on their device. Irrespective of whether the app is used now and then or not, the user count is representative of the audience who downloaded the app and found it worth retaining on their device. The total user count can further be divided into two categories - those who actively use the app and others who have kept the app as a decorative in their app drawer. A person who downloaded the app for the first time decided to keep it because he realized that it was worth installing. While he may not frequently use the app, it might come in handy at some point[1].

- Represents the company's purpose or core business.
  - Objective : Drive more customers to download our mobile app and register for the loyalty program.
  - Reason : User counts can measure how many interested customers have downloaded the mobile app on their devices.
- Simple to communicate with stakeholders.
  - Stakeholder : Internal team, manager, executives.
  - Reason : When using this metric, stakeholders can understand how far the mission/goal has been achieved.

Driver Metrics : Click-Through-Rate (CTR)

CTR is the ratio of the number of clicks on a specific link or an element of interface to the number of times people were exposed to the link or element[2]. To calculate CTR then :

$$\frac{\textit{people who click to download mobile app from website}}{\textit{total number of website visitor}} \textit{ x 100}$$

Guardrail Metrics : Mobile app loading time

Mobile app load time can be characterized as the measure of time taken by the application to totally introduce before the interface opens and the application gets significant or interactive for the user. Faster is always better when it comes to app loading time but 2 seconds is considered the acceptable limit for app loading time. According to Google, roughly half of users will abandon an app that takes three seconds or longer to load, higher abandonment rates occur as the loading time increases[3]. If mobile app loading time increases a few ms -> decreased satisfaction -> abandon/uninstall mobile app -> lose users -> potential loss.

3. **Define Variants**

   Control : Existing link.

   Treatment : New link, such as in picture not text.

4. **Define Hypothesis**

   H_0 (Null Hypothesis) : CTR New link such as in the picture, not text equal to or less than the existing link.

   H_1 (Alternative Hypothesis) : CTR New link such as in the picture, not text more than the existing link.

III. Designing Experiments

1. **Randomization Unit** : User
2. **Target of Randomization Unit** : All users who visit the web pages of the grocery store chain that contain links.
3. **Sample Size**

   For calculate sample size, we need to define :

   a. Significance level (α) is the probability that the experiment will produce a false-positive result (a type I or an error)[4]. This is the error and must be minimized. Therefore, the smaller the alpha the better for experiments but the larger the sample required. The significance level value in this experiment is 5% or 0.05. This is the most commonly used significance level/industry standard which indicates that the acceptable risk of error is 5% or 0.05. A significance level of 0.05 means that we accept a 5% chance of making a Type I error that rejects the true null hypothesis, in exchange for 95% confidence that the result is not due to chance. It was chosen because it provides a balance between the risk of type I error and type II error. Type I error is an error that occurs when rejecting H_0 (null hypothesis) when H_0 is true, while type II error occurs when failing to reject H_0 (null hypothesis) when H_0 is false.

On the other hand, a significance level of 0.01 is more conservative than a significance level of 0.05, conservative meaning when the desired experimental results are as accurate as possible.

b. Power level (1-β) is the probability that the experiment will reject the null hypothesis when it is false[4]. This is the truth that must be brought up. Therefore, the greater the power level the better for the experiment but the larger the sample required. The power level value in this experiment is 80% or 0.8. This is because a high power level value indicates that the experiment has a strong ability to detect significant differences between the tested groups if such differences do exist. The higher the power level value, the more likely it is that the experiment will be able to detect true differences between the groups. If type I errors are more important, then a higher power level value can be chosen to minimize the risk of such errors.
c. Standard deviation of population (σ) is a measure of how much variation there is among individual data points in a population. It's a way of quantifying how spread out the data is from its mean. A small standard deviation means that the data points are generally close to the mean, while a large standard deviation means that the data is more dispersed[5]. Since there is no historical data in this experiment, the value of the standard deviation of the population will be assumed to be 0.5.
d. Difference between control and treatment (δ) is minimum effect (difference) to be detected. The minimum detectable effect is the effect size set by the researcher that an impact evaluation is designed to estimate for a given level of significance. The minimum detectable effect is a critical input for power calculations and is closely related to power, sample size, and survey and project budgets[6]. MDE is generally expressed as a percentage or proportion of the difference between the groups being compared. For example, if we want to detect a difference in conversions between two groups with a 95% significance level, we might decide that the MDE that can be detected is 5%. That is, if the difference between the two groups is less than 5%, we may not be able to significantly distinguish between the two in the experiment. In this experiment, the difference between the control and treatment is 2%. In this case, we want to be able to detect a difference of 2%.
   Then the number of Sample Size :

$$n \approx \frac{16\sigma^2}{\delta^2}$$

   n = Number of samples
   σ = Standard deviation of population
   δ = Difference between control and treatment

$$n \approx \frac{16(0.5)^2}{0.02^2} = 10.000$$

   Sample size 10.000 for 1 variant, so total for 2 variants : 10.000 x 2 = 20.000.
e. Since this experiment requires a very large sample size, the length of time to run the experiment depends on the number of visitors to the website. If the experiment is run for 6 full weeks with the frequency of users visiting the website at least 500 times per day, then the total number of users involved in the experiment is

42 days x 500 = 21,000. From this, the time sufficient to collect data is at least 6-8 weeks, the length of this experiment is done to avoid primacy and novelty effects. The primacy and novelty effect occurs when the initial interest is high, but as time goes by it drops.

IV. Analyzing and Interpreting the Data

The dataset used in this project comes from Grocery website data for AB test. The dataset has 184.588 records with 5 variables. Here is information from the data variables used:

- RecordID : identifier of the row of data.
- IP Address : address of the user, who is visiting website.
- LoggedInFlag : 1 - when user has an account and logged in.
- ServerID : one of the servers user was routed through.
- VisitPageFlag : 1 - when user clicked on the loyalty program page.

From the sample size calculation, 10,000 users are obtained for each variant. Therefore, Simple Random Sampling is carried out to get a sample. To analyze and interpret data, the following steps were taken:

1. Ensure the trustworthiness
   a. Check the data quality (missing value, duplicate data, distribution of data).

```
#checking missing value
data.isna().sum()
```
✓ 0.3s Python

```
RecordID        0
IP Address      0
LoggedInFlag    0
ServerID        0
VisitPageFlag   0
dtype: int64
```

There is no missing value. Next, check for duplicate data.

```
#checking duplicate data
data.duplicated(['IP Address']).sum()
```
✓ 0.4s Python

```
85072
```

There are 85,072 duplicate data, so delete the duplicate data.

```
#droping duplicates data
data.drop_duplicates(subset='IP Address', inplace=True)
data.shape
```
✓ 0.5s Python

```
(99516, 5)
```

Now, there are 99.516 total records without duplicates and the data is ready for analysis.

b. Data exploration (how many users in each group, and other insight from dataset)
Each user in the control group and the treatment group has 10,000 users as shown below.

```
#choose sample on control
sample_data_control = data[data['Group']=='Control'].sample(n = n,
                                                            replace = False)
#show sample on control
sample_data_control
```

✓ 0.2s Python

	RecordID	IP Address	LoggedInFlag	ServerID	VisitPageFlag	Group
20465	20466	109.9.233.9	1	3	0	Control
128191	128192	19.10.120.4	1	2	0	Control
1022	1023	254.1.172.4	0	3	0	Control
99161	99162	214.13.250.8	0	2	0	Control
28765	28766	81.15.243.6	1	2	0	Control
...	...	...	...	...	...	...
143953	143954	217.9.122.3	1	2	0	Control
24741	24742	16.16.179.2	0	2	0	Control
55141	55142	150.9.108.7	0	2	0	Control
17077	17078	10.12.146.9	1	2	0	Control
4460	4461	46.12.32.3	1	2	0	Control

10000 rows × 6 columns

```
#choose sample on treatment
sample_data_treatment = data[data['Group']=='Treatment'].sample(n = n,
                                                            replace = False)
#show sample on treatment
sample_data_treatment
```

✓ 0.2s Python

	RecordID	IP Address	LoggedInFlag	ServerID	VisitPageFlag	Group
125082	125083	83.5.181.6	0	1	1	Treatment
167063	167064	147.0.161.8	0	1	0	Treatment
87565	87566	134.0.112.5	1	1	0	Treatment
140719	140720	38.3.237.5	1	1	0	Treatment
75321	75322	165.4.84.4	0	1	1	Treatment
...	...	...	...	...	...	...
24019	24020	208.16.198.6	1	1	0	Treatment
119602	119603	82.11.205.9	0	1	0	Treatment
32568	32569	126.15.64.7	0	1	0	Treatment
113858	113859	254.12.206.4	1	1	0	Treatment
153921	153922	30.4.114.9	0	1	0	Treatment

10000 rows × 6 columns

Then calculate the CTR, for both groups as follows.

```
#find control group that is CTR
data_control_ctr = sample_data_control[(sample_data_control['VisitPageFlag'] == 1)]

#length of CTR on control group
n_control_ctr = len(data_control_ctr)
n_control_ctr
```

✓ 0.1s Python

513

```
#find treatment group that is CTR
data_treatment_ctr = sample_data_treatment[(sample_data_treatment['VisitPageFlag'] == 1)]

#length of CTR on treatment group
n_treatment_ctr = len(data_treatment_ctr)
n_treatment_ctr
```

✓ 0.1s Python

691

To see more clearly, compare the control group and the treatment group. Create the following code.

```
#merge sample data control and treatment
data_sample = pd.concat([sample_data_control, sample_data_treatment], ignore_index=True)

#comparing the Control and Treatment Group on VisitPageFlag
group_with_visitPage = pd.crosstab(data_sample['Group'], data_sample['VisitPageFlag'], margins=True)
group_with_visitPage
```

✓ 0.3s Python

VisitPageFlag	0	1	All
Group			
Control	9487	513	10000
Treatment	9309	691	10000
All	18796	1204	20000

Next make a visualization, to see the comparison of CTR on each variant.

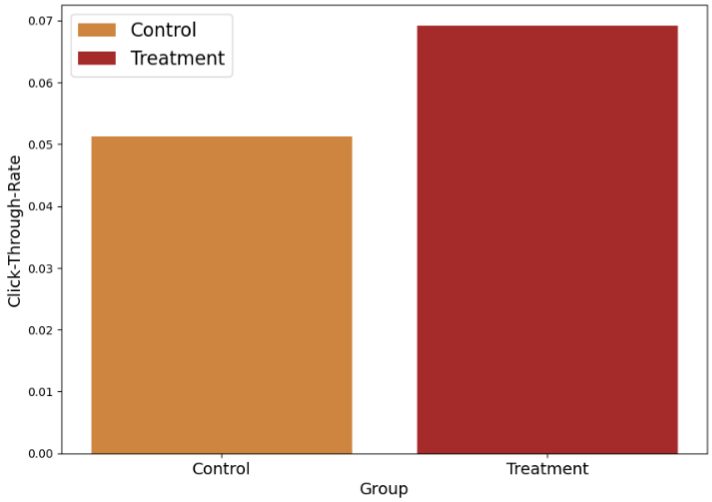
```
#compare visualization CTR each variant
fig, ax = plt.subplots(nrows=1, ncols=1, figsize=(10,7))

#make plot
ax.bar("Control", ctr_control, color="peru", label="Control")
ax.bar("Treatment", ctr_treatment, color="brown", label="Treatment")

#styling plot
ax.set_ylabel("Click-Through-Rate", fontsize=14)
ax.set_xlabel("Group", fontsize=14)
ax.set_xticklabels(labels=["Control", "Treatment"], fontsize=14)
ax.legend(fontsize=16)
plt.show()
```

Python

```
C:\Users\UYUN\AppData\Local\Temp\ipykernel_13972\3857991313.py:11: UserWarning: FixedFormatter should only be used together with FixedLocator
  ax.set_xticklabels(labels=["Control", "Treatment"], fontsize=14)
```



c. Perform SRM test with chi-square test

- Define the null and alternative hypothesis (H_0 and H_1)

  H_0 : No SRM detected

  H_1 : SRM detected

- Calculate chi-square statistics

$$\chi^2 = \sum \frac{(\text{observed - expected})^2}{\text{expected}}$$

Where :
- Observed: the control and variation traffic volumes (sample size), respectively.
- Expected: the expected values for control and treatment — i.e. the total observed divided by 2.

- Define decision rules

In making statistical test decisions, we can use:
- Comparison of chi-square statistics with critical value

$$\chi^2 > \chi^2_{\alpha,df} \rightarrow \text{reject } H_0$$

- Comparison of p-value with alpha

$$\text{pvalue} < \alpha \rightarrow \text{reject } H_0$$

- Degree of freedom (df) is calculated as:

$$df = (rows - 1) \times (columns - 1)$$

```python
# Comparison of chi-square statistics with critical value
# We must calculate the critical first

# critical value is the chi-square value at alpha
alpha = 0.05
df=(2-1)*(2-1)

import scipy
chi_critical = scipy.stats.chi2.ppf(1 - alpha, df)
print(f"Critical value: {chi_critical:.3f}")
```

Python

```
Critical value: 3.841
```

```python
#Make decisions from chi-square statistics and critical value
if chi[0] > chi_critical:
  print("Reject H0 : SRM may be present.")
else:
  print("Fail to reject H0 : No SRM")
```

Python

```
Fail to reject H0 : No SRM
```

```python
# Comparison of P-Value with alpha.
if chi[1] < 0.01:
    print('Reject H0 : SRM may be present.')
else:
    print('Fail to reject H0 : No SRM.')
```

Python

```
Fail to reject H0 : No SRM.
```

Based on the detection of SRM, SRM was not detected.

2. Conduct hypothesis testing and analyze the result

- Define null hypothesis H_0 and alternative hypothesis H_1

H_0 (Null Hypothesis) : CTR New link, such as in picture not text ≤ existing link

H_1 (Alternative Hypothesis) : CTR New link, such as in picture not text > existing link

First, define Z_{crit}, $Z_{statistic}$, and p-value. To calculate $Z_{statistic}$ and p-value use this function.

```python
# Import this library to calculate
import statsmodels.api as sm
from statsmodels.stats.proportion import proportions_ztest
```

Python

```python
# Make count convert & total observation
count_convert = [n_treatment_ctr, n_control_ctr]
count_observation = [n_treatment, n_control]
```

Python

Create an alternative for this hypothesis test case, in this case use 'larger' because want to prove CR_{new} is greater than CR_{old}.

```python
#make alternative
alternative_option = "larger"
```

Python

```python
#call function
z_stat, p_value = proportions_ztest(count = count_convert,
                                    nobs = count_observation,
                                    alternative = alternative_option)

print(f"Z stats : {z_stat:.4f}")
print(f"P-value : {p_value}")
```

Python

```
Z stats : 5.2916
P-value : 6.061722707735026e-08
```

Then calculate the difference between the new and old CR.

```python
ctr_treatment - ctr_control
```

Python

```
0.017799999999999996
```

```python
# addition
# we can calculate the relative effect which shows how much the percentage increase or decrease in the CTR treatment compared to the control
relative_effect_CTR = (ctr_treatment - ctr_control)/ctr_control * 100

print(f"relatife effect = {relative_effect_CTR:.3} %")
```

Python

```
relatife effect = 34.7 %
```

There is a relative increase of 34.7%. Next, summarize the statistical test results.

```python
# from p-value
alpha = 0.05        # the confidence level that we set

# make a decision based on p_value and alpha
if p_value < alpha:
    print("Decision : Reject Null Hypothesis")
else:
    print("Decision : Fail to Reject Null Hypothesis")
```

Python

```
Decision : Reject Null Hypothesis
```

```python
# Z critical is the z-value at alpha
z_critical = stats.norm.ppf(1 - alpha)
z_critical
```

Python

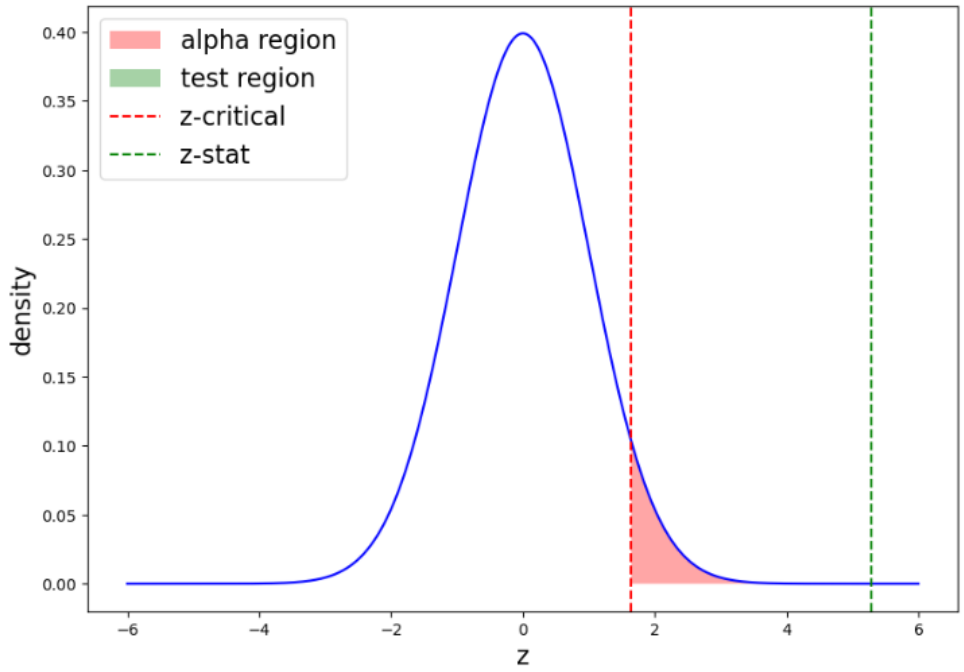
```
1.6448536269514722
```

```python
# from z-statistics
# make decision based on z_critical and alpha
if z_stat > z_critical:
    print("Decision : Reject Null Hypothesis")
else:
    print("Decision : Fail to Reject Null Hypothesis")
```

Python

```
Decision : Reject Null Hypothesis
```

Next, visualize the statistical test results above. The visualization is made in a z value distribution graph. Therefore, find the z value when alpha = 0.05. The results of the visualization obtained will be seen in the following figure.



From the visualization above, the area of the green region < the area of the red region (the region where H_0 is rejected) means that the probability of getting H_0 from the sample is even smaller than the set alpha limit. Statistically, there is not enough evidence to accept H_0 (the p-value is smaller than alpha), so H_0 is rejected.

3. Calculate confidence interval of difference between treatment and control

```
from statsmodels.stats.proportion import confint_proportions_2indep

confidence_interval = confint_proportions_2indep(count1 = n_treatment_ctr, nobs1 = n_treatment,
                                                 count2 = n_control_ctr, nobs2 = n_control,
                                                 compare='diff', alpha=0.05)
print(confidence_interval)
```

Python

```
(0.011216025374711922, 0.02440743382202487)
```

Based on this result, there is 95% confidence that the difference in the proportion of users who clicked on the new link (CTR) in the treatment (B) and control (A) groups is between 0.011 and 0.024. Or it can be said that the increase in users when downloading applications using new links such as in picture (not text) (treatment) increases by 0.011 to 0.024.

V. Conclusion and Recommendation

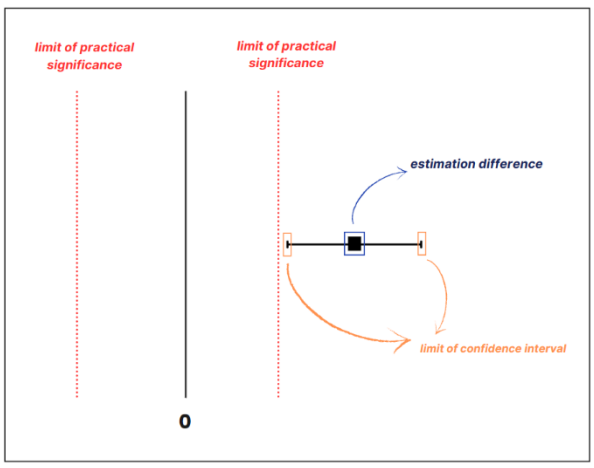
- P-value (6.061722707735026e-08) < α (0.05) -> Reject H0
- Z Statistic (5.2916) > Z Critical (1.644) -> Reject H0

With significance level 5%, there is sufficient evidence that CTR New link such as in picture not text (treatment) more than existing link (control). In other words, CTR New link, such as in picture not text will increase user interest in downloading the application.

Recommendations for website of the grocery store chain :

- Based on the statistical test results, the results are statistically significant. P-value = 0.05 indicates that there is a 5% probability that the observed difference is due to chance or other factors unrelated to the variable being observed.
- But to make a decision whether to change the link to a button of the app store or not, must be practically significant such as :
  1. Resources and costs required to implement the change. If the cost required for the change to the link to a button of the app store on the website is very high and not proportional to the impact on mobile app downloads, then the change may not be considered practically significant.
  2. It is also necessary to consider the difference between performance before and after the change. If the change to the link to a button of the app store on the website can increase mobile app downloads by 1% or more, then the change may be considered practically significant. However, if the change only increases mobile app downloads by 0.1% or less, then the change may not be considered practically significant.
- Based on the above considerations, the change is considered practically significant.

launch feature ->



Recommendation for the next experiment :

1. Download page variants: change the layout or content of the mobile app download page, such as adding images or positive reviews from other users.
2. App description: Change the app description on the website, such as highlighting the benefits or advantages of the app.
3. Changes to the overall appearance and content of the website: Changing the overall layout, design, and content of the website.
4. Target audience: There may be certain groups of users who are more likely to download apps than others, so changing the look and content of the website to appeal more to certain target groups could be a recommendation for future experiments.

VI. References

1. mobileappdaily.com, Top 8 App Engagement Metrics For Mobile Apps To Track in 2023. March 14, 2023. [Accessed on April 1, 2023]. https://www.mobileappdaily.com/top-metrics-to-measure-user-engagement.
2. Damaševi˘cius Robertas, Zailskaite-Jakšte Ligita. Usability and Security Testing of Online Links: A Framework for Click-Through Rate Prediction Using Deep Learning, 2022.
3. storyly.io, App Loading. [Accessed on April 1, 2023]. https://www.storyly.io/glossary/app-loading.
4. Festing Michael FW. On determining sample size in experiments involving laboratory animals, 2017.
5. Khanacademy.org, Population standard deviation. [Accessed on April 1, 2023]. https://www.khanacademy.org/math/statistics-probability/summarizing-quantitative-data/variance-standard-deviation-population/v/population-standard-deviation#:~:text=The%20population%20standard%20deviation%20is,data%20is%20from%20its%20mean.
6. Dimewiki.worldbank.org, Minimum Detectable Effect. [Accessed on April 9, 2023]. https://dimewiki.worldbank.org/Minimum_Detectable_Effect#:~:text=The%20minimum%20detectable%20effect%20is,and%20survey%20and%20project%20budgets.