Published in Towards Data Science

Terence Shin Follow
Jun 1, 2020 · 7 min read ★ · Listen

A/B Testing with Chi-Squared Test to Maximize Conversions and CTRs

A Data Science Project Walkthrough for Aspiring Experimentalists

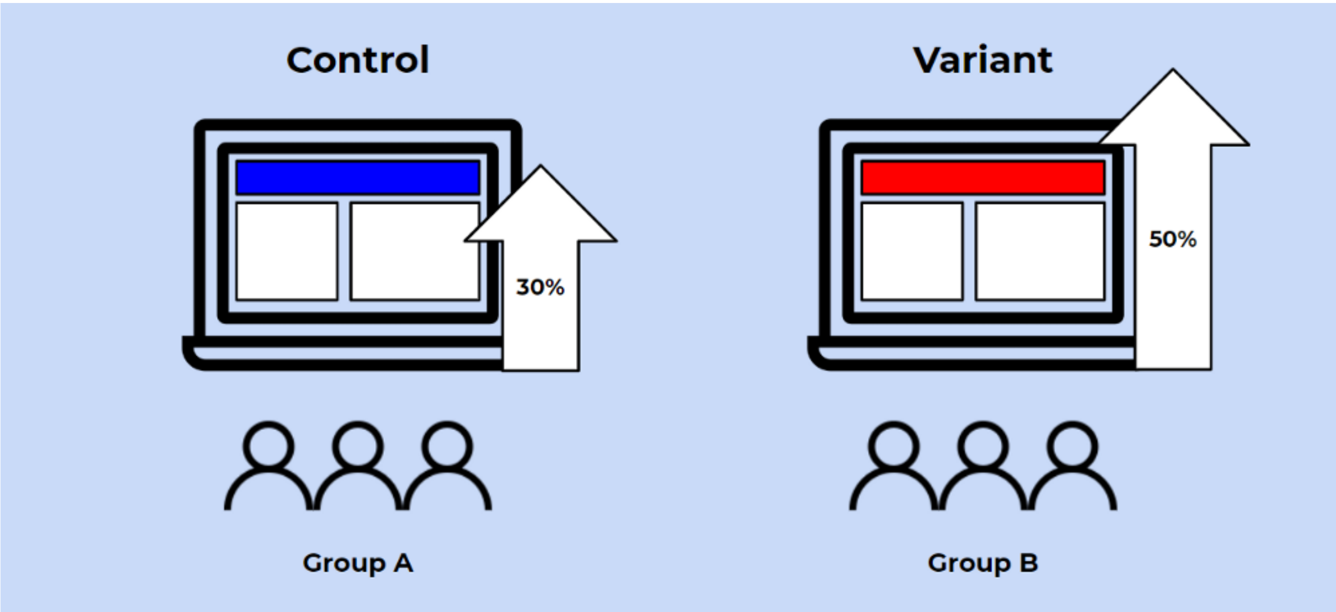


Image created by author

Table of Content

The first four sections explain the concepts that are used for this A/B test and the last two sections go over a step-by-step example of performing an A/B test on conversion rates.

Introduction

Arguably one of the most practical data science concepts in the workplace is A/B Testing. And yet, it is a concept that is quite misunderstood because there are a lot of intricacies to it.

To give an example, many experimentalists use the t-test to determine if there are significant differences between the two alternatives. But what if the distribution is not assumed to be Gaussian? What if the standard deviations of the two groups are different? What if the distribution is completed unknown?

In this article, I'm going to talk about a particular A/B Testing method that works well for comparing click-through rates and conversions.

What is an A/B Test?

A/B testing in its simplest sense is an experiment on two variants to see which performs better based on a given metric. Typically, two consumer groups are exposed to two different versions of the same thing to see if there is a significant difference in metrics like sessions, click-through rate, and/or conversions.

Using the visual above as an example, we could randomly split our customer base into two groups, a control group and a variant (or treatment) group. Then, we can expose our variant group with a red website banner and see if we get a significant increase in conversions. It's important to note that all other variables need to be held constant compared against the population data. **Two-sample hypothesis testing** is a method in determining whether the differences between the two samples are statistically significant or not.

Chi-Squared Test

What do click-through rates and conversions have in common? They have a **Bernoulli Distribution**, the discrete probability distribution that has a probability of being 1 and a probability of being 0. For click-through rates, a user will either click (1) or not click (0). Similarly, for conversions, a user will either convert (1) or not convert (0).

Because we're performing an A/B Test on conversions which is a categorical variable that follows a Bernoulli distribution, we'll be using the **Chi-Squared Test**.

The steps to conducting a chi-squared test are as follows:

1. Calculate the chi-squared test statistic
2. Calculate the p-value
3. Compare the p-value against the level of significance

This will make more sense when you follow along with the project walkthrough.

Calculating the Chi-Squared Test Statistic

The equation to determine the chi-squared test statistic is as follows:

$$x^2 = \sum_i \frac{(observed_i - expected_i)^2}{expected_i}$$

Imagine we showed two different advertisements, A and B, to test whether users clicked or didn't click the advertisement. At the end of the test, the following information was collected:

	Click	No Click	Click + No Click
Advertisement A	360	140	500
Advertisement B	300	250	550
Ad A + Ad B	660	390	1050

In this case, we would have to make four calculations and then sum them:

1. Advertisement A, Click
2. Advertisement A, No Click
3. Advertisement B, Click
4. Advertisement B, No Click

Let's use advertisement A as an example. We would need to calculate the observed value and the expected value.

The observed value is equal to 360, as shown in the table above.

The expected value is equal to the number of times ad A is shown multiplied by the probability of a click. Thus, expected value = 500 * (660/1050) = 31.429. These numbers are taken from the table above as well.

After calculating this for all four scenarios, the numbers can be inputted into the x^2 equation to determine the chi-squared test statistic.

The dataset that I used to perform this A/B test is taken from Kaggle (here is the link to the data set).

This dataset contains the result of an A/B test where two groups, the control group and the treatment group, were exposed to an old webpage and a new webpage respectively. The purpose of this test was to determine if the new webpage resulted in a significant increase in conversions compared to the old webpage. Each row represents a unique user and shows whether they're in the control or treatment group and whether they converted or not.

A/B Test Project Walkthrough

```
# Import libraries and data
import numpy as np
import pandas as pd
import scipy
import matplotlib.pyplot as plt

df = pd.read_csv('../input/ab-testing/ab_data.csv')
```

As always, I started off by importing the relevant libraries and the data.

Data Wrangling

Before performing the chi-squared test, I wanted to check a couple of things since I didn't know how clean the data was. The first thing I checked was to see if there were any users in the control group that saw the new web page and vice versa.

```
# Checking to see if there are any users in control that saw new
page and users in treatment that saw old page
df.groupby(['group','landing_page']).count()
```

		user_id	timestamp	converted
group	landing_page			
control	new_page	1928	1928	1928
	old_page	145274	145274	145274
treatment	new_page	145311	145311	145311
	old_page	1965	1965	1965

Above, you can see that there seems to be an error where some users in the control group saw the new page and some users in the treatment group saw the old page. Since I wasn't sure which way to revert the misclassified users, I decided to remove it with the code below.

```
# Removing control/new_page and treatment/old_page
df_cleaned = df.loc[(df['group'] == 'control') & (df['landing_page']
== 'old_page') | (df['group'] == 'treatment') & (df['landing_page']
== 'new_page')]

df_cleaned.groupby(['group','landing_page']).count()
```

		user_id	timestamp	converted
group	landing_page			
control	old_page	145274	145274	145274
treatment	new_page	145311	145311	145311

Now, the control group is restricted to the old page and the treatment group is restricted to the new page. The next thing that I wanted to check for was duplicate values based on user_id.

```
# Checking for duplicate values
df_cleaned['user_id'].duplicated().sum()
```

```
1
```

```
# Finding user_id for duplicate value
df_cleaned[df_cleaned.duplicated(['user_id'],keep=False)]['user_id']
```

```
1899    773192
2893    773192
Name: user_id, dtype: int64
```

```
df[df['user_id'] == 773192]
```

	user_id	timestamp	group	landing_page	converted
1899	773192	2017-01-09 05:37:58.781806	treatment	new_page	0
2893	773192	2017-01-14 02:55:59.590927	treatment	new_page	0

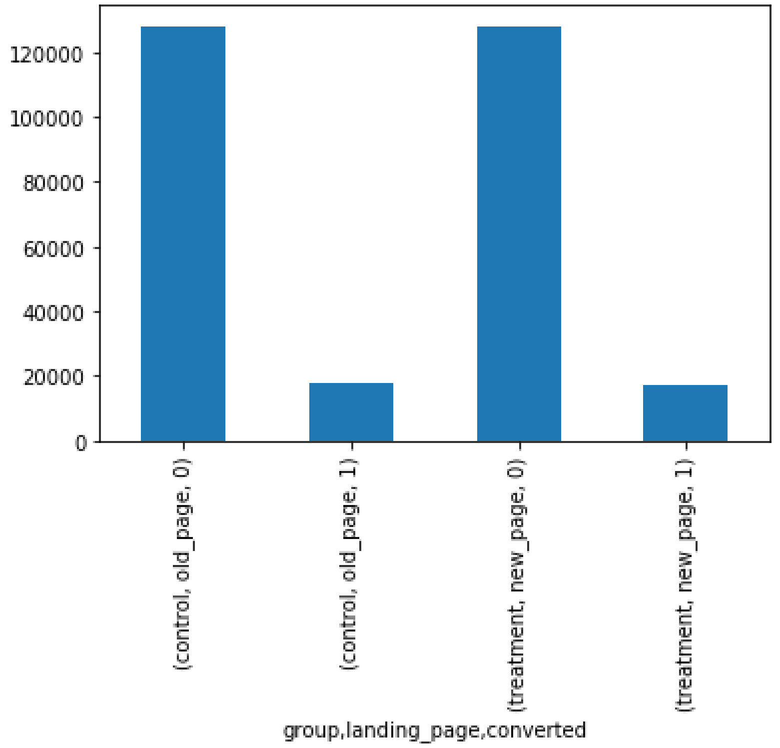
It looked like this user saw the new landing page twice and didn't convert both times. For the simplicity of the experiment, I wanted to restrict it to each user's first decision only. Thus, I removed the second instance for this user.

```
df_cleaned = df.drop_duplicates(subset='user_id', keep="first")
```

Exploratory Data Analysis

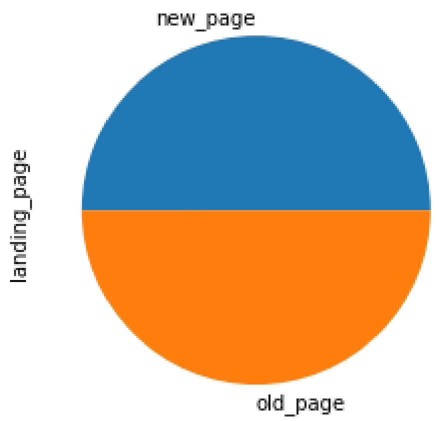
Once my data was cleaned, I wanted to get a better understanding of my data. I plotted the data against a bar chart to see what the proportion of conversions was for both groups. It appears that they have similar conversion rates (approx. 1/7), but we'll see whether there's a significant difference or not through the chi-squared test.

```
groups =
df_cleaned.groupby(['group','landing_page','converted']).size()
groups.plot.bar()
```



As well, I wanted to see if the proportion between the number of users in each group were similar in size using a pie chart.

```
df_cleaned['landing_page'].value_counts().plot.pie()
```



Data Preparation

```
### Re-arrrange data into 2x2 for Chi-Squared

# 1) Split groups into two separate DataFrames
a = df_cleaned[df_cleaned['group'] == 'control']
b = df_cleaned[df_cleaned['group'] == 'treatment']

# 2) A-click, A-noclick, B-click, B-noclick
a_click = a.converted.sum()
a_noclick = a.converted.size - a.converted.sum()
b_click = b.converted.sum()
b_noclick = b.converted.size - b.converted.sum()

# 3) Create np array
T = np.array([[a_click, a_noclick], [b_click, b_noclick]])
```

Chi-Squared Test

Once my data was in the proper format, I was ready to conduct the test. This can simply be done by importing stats from the Scipy library. This step calculates both the chi-squared statistic and the p-value.

```
import scipy
from scipy import stats

print(scipy.stats.chi2_contingency(T,correction=False)[1])
```

```
0.18988337448194853
```

The p-value was calculated to be 19%. Assuming a 5% level of significance, we can deduce that the p-value is greater than the alpha and that we do not reject the null hypothesis. In simpler terms, there is no significance in conversions between the old and new webpage.

```
# Sanity Check
a_CTR = a_click / (a_click + a_noclick)
b_CTR = b_click / (b_click + b_noclick)
print(a_CTR, b_CTR)
```

As a sanity check, I calculated the conversion rates between the two groups and the difference between them is minimal, which reassures that this was conducted properly.

And that's the end of this tutorial! I hope you found this useful. :)

Thanks for Reading!

Terence Shin

Founder of ShinTwin | Let's connect on LinkedIn | Project Portfolio is here.