

# ab\_testing

September 18, 2021

## 1 A/B Testing

This is a demonstration of new website strategies with a hypothetical company called **Mattresses and More**.

Their strategy is using modal pop-ups to show features of the site, such as: - Welcome to Mattresses and More! - Browse our bedding selection - Select items to add to your cart - View your cart by clicking on the icon - Press 'Buy Now!' when you're ready to checkout

### 1.1 Objective and Design

Our product team wants to create a new design for the pop-ups that they believe will be more amicable to the user base.

We've set up an A/B test where: - 50% of users view the original **control** version of the pop-ups - 50% of users view the new **variant** version of the pop-ups

We will be using a table called **onboarding\_modals** with the following columns: - **user\_id**: the user identifier - **modal\_text**: the modal step - **user\_action**: the user response (Close Modal or Continue) - **ab\_group**: the version (control or variant)

### 1.2 Imports

We'll be using **sqlite3** for Python for our SQL handling in this format.

```
[1]: import sqlite3
```

### 1.3 SQL

Let's first connect to our database which we've already created in **make\_db.py** from **onboarding\_modals.txt**.

```
[2]: # creates the database or connects to it
conn = sqlite3.connect('onboarding_modals.db')

# creates a cursor for sql commands
cur = conn.cursor()
```

### 1.3.1 Exploration

Now we can explore our database. Let's look at the first 10 entries.

```
[3]: cur.execute("""SELECT *
                FROM onboarding_modals
                LIMIT 10;
            """)

print(cur.fetchall())
```

```
[('0015585f-51d0-4654-83fc-acce6544b0cf', '1 - Welcome to Mattresses and More!',
'Close Modal', 'control\n'), ('0028b8b4-abb3-4711-9ea2-0d1ecb975358', '1 -
Welcome to Mattresses and More!', 'Close Modal', 'control\n'),
('0029802c-22a1-4d22-9a59-244341551ec9', '1 - Welcome to Mattresses and More!',
'Continue', 'variant\n'), ('0029802c-22a1-4d22-9a59-244341551ec9', '2 - Browse
our bedding selection', 'Continue', 'variant\n'),
('0029802c-22a1-4d22-9a59-244341551ec9', '3 - Select items to add to your card',
'Continue', 'variant\n'), ('0029802c-22a1-4d22-9a59-244341551ec9', '4 - View
your cart by clicking on the icon', 'Continue', 'variant\n'),
('0029802c-22a1-4d22-9a59-244341551ec9', '5 - Press 'Buy Now!' when you're ready
to checkout', 'Continue', 'variant\n'), ('0042ec6f-e343-4486-a009-ecaea4ff31b2',
'1 - Welcome to Mattresses and More!', 'Continue', 'variant\n'),
('0042ec6f-e343-4486-a009-ecaea4ff31b2', '2 - Browse our bedding selection',
'Close Modal', 'variant\n'), ('0121762e-7d4e-46b0-94d8-7bc56a5cb88b', '1 -
Welcome to Mattresses and More!', 'Close Modal', 'control\n')]
```

### 1.3.2 Funnel Completion

In order to see how well our funnel performs, we can track the progress through the `modal_text` and group by how many unique `user_id` completed the funnel. We can also sort the output by question in the funnel to make it easier to see the progression.

```
[4]: cur.execute("""SELECT modal_text, COUNT(DISTINCT user_id)
                FROM onboarding_modals
                GROUP by 1
                ORDER by 1;
            """)

print(cur.fetchall())
```

```
[('1 - Welcome to Mattresses and More!', 1000), ('2 - Browse our bedding
selection', 695), ('3 - Select items to add to your card', 575), ('4 - View your
cart by clicking on the icon', 447), ('5 - Press 'Buy Now!' when you're ready to
checkout", 379)]
```

It's pretty clear that most people don't make it all the way through the funnel: - Modal 1: 1000 - Modal 2: 695 - Modal 3: 575 - Modal 4: 447 - Modal 5: 379

### 1.3.3 A/B Testing

Since we have our groups marked as **control** and **variant**, we can compare how successful our individual groups were at making it through the funnel!

```
[5]: cur.execute("""SELECT modal_text,
                    COUNT(DISTINCT CASE WHEN ab_group LIKE 'control%' THEN
↪user_id
                    END) AS 'control_clicks',
                    COUNT(DISTINCT CASE WHEN ab_group LIKE 'variant%' THEN
↪user_id
                    END) AS 'variant_clicks'
FROM onboarding_modals
GROUP BY 1
ORDER BY 1;

""")

print(cur.fetchall())
```

```
[('1 - Welcome to Mattresses and More!', 500, 500), ('2 - Browse our bedding
selection', 301, 394), ('3 - Select items to add to your card', 239, 336), ('4 -
View your cart by clicking on the icon', 183, 264), ('5 - Press 'Buy Now!' when
you're ready to checkout", 152, 227)]
```

Since we have the data we need we can close the connection.

```
[6]: # close connection
conn.close()
```

### 1.3.4 Results

Great! It looks like the **variant** of our modal test was successful!

In terms of user progression through the funnel:

Modal Funnel	Control	Variant
1 - Welcome to Mattresses and More!	500	500
2 - Browse our bedding selection	301	394
3 - Select items to add to your card	239	336
4 - View your cart by clicking on the icon	183	264
5 - Press 'Buy Now!' when you're ready to checkout	152	227

## 1.4 Conclusion

In this project, we performed A/B Testing of a hypothetical mattress company's strategy to push users from site entry to checkout. After performing the test we found that our **variant** of the questions did indeed push more users into checkout.

## 1.5 Further Research

We could take this further by performing statistical analysis of the difference between the modals and determine if it is significant enough for our team to change the modal questions. Additionally, we could explore the elimination of the modal pop-up in comparison to other strategies to push users through the funnel.

For more complex datasets and analysis we could either perform more complex SQL queries or turn this database into a dataframe which could make some analyses easier to perform.

**Data Sources** Data was provided by [Codecademy.com](https://www.codecademy.com).