A/B testing for marketing

ANALYZING MARKETING CAMPAIGNS WITH PANDAS

Jill Rosok
Data Scientist

What is A/B testing?

Prior to running the test determine:

- What is the desired outcome of the test? What is our hypothesis?
- What is the metric we are trying to impact (i.e., page views, conversions)?
- Will we get enough traffic to our site to reach statistical significance and make a decision in a timely manner?

Testing allows us to understand marketing impact

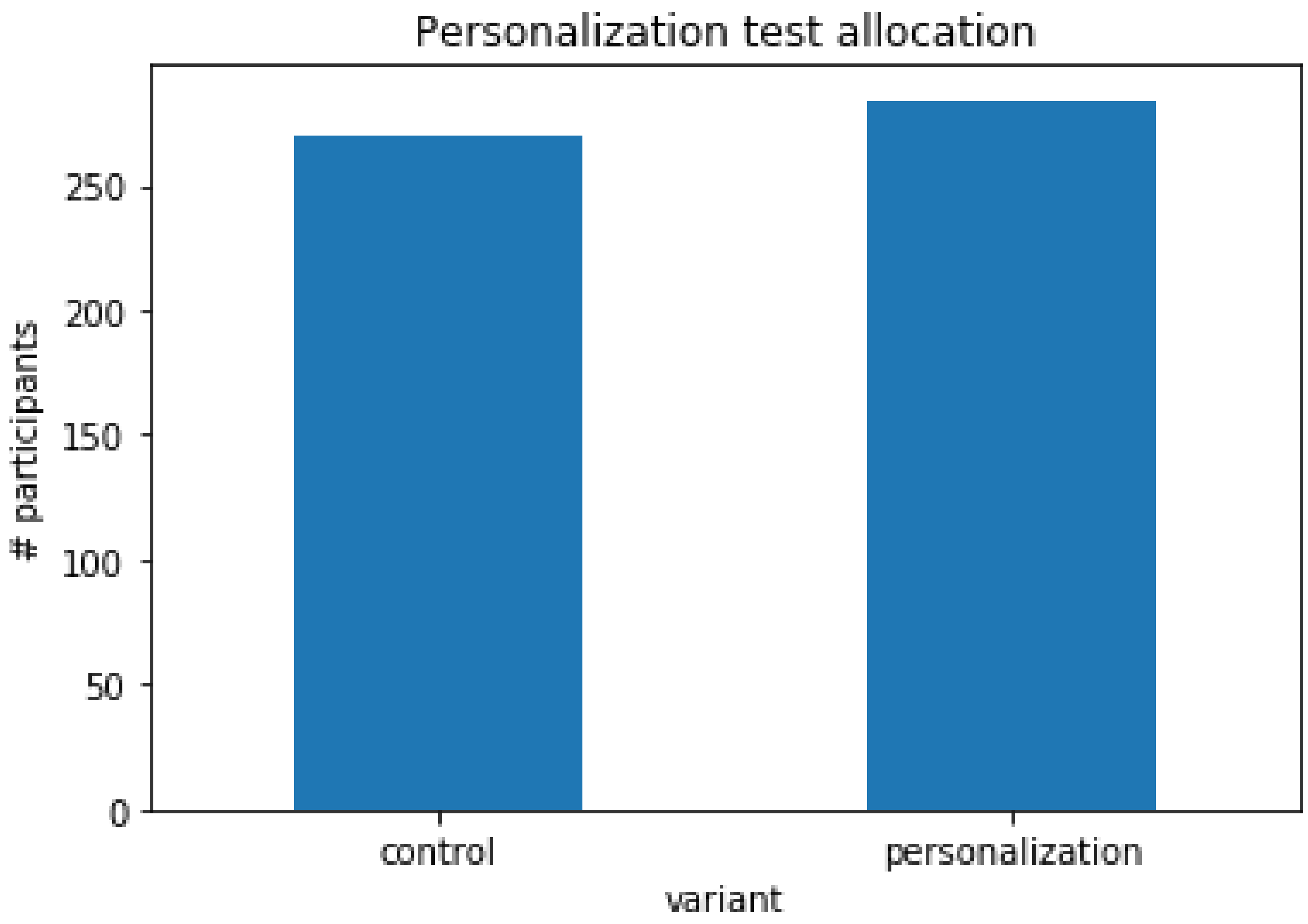
How long does a test need to run?

Personalized email test

Test allocation

```
email = marketing[marketing['marketing_channel'] == 'Email']
allocation = email.groupby(['variant'])\
                           ['user_id'].nunique()
allocation.plot(kind='bar')
plt.title('Personalization test allocation')
plt.xticks(rotation = 0)
plt.ylabel('# participants')
plt.show()
```

Allocation plot



Setting up our data to evaluate the test

```python
# Group by user_id and variant
subscribers = email.groupby(['user_id',
                             'variant'])['converted'].max()


subscribers = pd.DataFrame(subscribers.unstack(level=1))
```

Setting up our data to evaluate the test

```
# Drop missing values from the control column
control = subscribers['control'].dropna()

# Drop missing values from the personalization column
personalization = subscribers['personalization'].dropna()
```

Conversion rates

```
print("Control conversion rate:",
      np.mean(control))
print("Personalization conversion rate:",
      np.mean(personalization))
```

```
Control conversion rate: 0.2814814814814815
Personalization conversion rate: 0.3908450704225352
```

Let's get testing!

Calculating lift & significance testing

ANALYZING MARKETING CAMPAIGNS WITH PANDAS

Jill Rosok
Data Scientist

Treatment performance compared to the control

Calculating lift:

$$\frac{\text{Treatment conversion rate - Control conversion rate}}{\text{Control conversion rate}}$$

Calculating lift

```
# Calcuate the mean of a and b
a_mean = np.mean(control)
b_mean = np.mean(personalization)

# Calculate the lift using a_mean and b_mean
lift = (b_mean-a_mean)/a_mean

print("lift:", str(round(lift*100, 2)) + '%')
```

```
lift: 194.23%
```

T-distribution

overlapping t distributions

[1] Identification of Timed Behavior Models for Diagnosis in Production Systems. Scientific Figure on ResearchGate.

P-values

- T-statistic of 1.96 is typically statistically significant at the 95% level
- Depending on the context of the test, you may be comfortable with a lower or higher level of statistical significance.

T-test in Python

```python
from scipy.stats import ttest_ind

t = ttest_ind(control, personalized)

print(t)
```

```
Ttest_indResult(statistic=-2.7343299447505074,
                pvalue=0.006451487844694175)
```

Let's practice!

ANALYZING MARKETING CAMPAIGNS WITH PANDAS

A/B testing & segmentation

ANALYZING MARKETING CAMPAIGNS WITH PANDAS

Jill Rosok
Data Scientist

Don't forget about segmentation!

Personalization test segmented by language

```
for language in np.unique(marketing['language_displayed'].values):
    print(language)
```

Isolate the relevant data

```python
for language in np.unique(marketing['language_displayed'].values):
    print(language)

    language_data = marketing[(marketing['marketing_channel'] == 'Email') &
                              (marketing['language_displayed'] == language)]
```

Isolate subscribers

```
for language in np.unique(marketing['language_displayed'].values):
    print(language)

    language_data = marketing[(marketing['marketing_channel'] == 'Email') &
                              (marketing['language_displayed'] == language)]

    subscribers = language_data.groupby(['user_id', 'variant'])['converted']\
                                                                      .max()
```

Isolate control and personalization

```
for language in np.unique(marketing['language_displayed'].values):
    print(language)

    language_data = marketing[(marketing['marketing_channel'] == 'Email') &
                              (marketing['language_displayed'] == language)]

    subscribers = language_data.groupby(['user_id', 'variant'])['converted']\
                                                                      .max()
    subscribers = pd.DataFrame(subscribers.unstack(level=1))
    control = subscribers['control'].dropna()
    personalization = subscribers['personalization'].dropna()
```

Full for loop

```
for language in np.unique(marketing['language_displayed'].values):
    print(language)

    language_data = marketing[(marketing['marketing_channel'] == 'Email') &
                              (marketing['language_displayed'] == language)]

    subscribers = language_data.groupby(['user_id', 'variant'])['converted']\
                                                                      .max()
    subscribers = pd.DataFrame(subscribers.unstack(level=1))
    control = subscribers['control'].dropna()
    personalization = subscribers['personalization'].dropna()

    print('lift:', lift(control, personalization))
    print('t-statistic:', stats.ttest_ind(control, personalization), '\n\n')
```

Results

```
Arabic
lift: 50.0%
t-statistic: Ttest_indResult(statistic=-0.58, pvalue=0.58)

English
lift: 39.0%
t-statistic: Ttest_indResult(statistic=-2.22, pvalue=0.03)

German
lift: -1.62%
t-statistic: Ttest_indResult(statistic=0.19, pvalue=0.85)

Spanish
lift: 166.67%
t-statistic: Ttest_indResult(statistic=-2.36, pvalue=0.04)
```

Let's practice!

ANALYZING MARKETING CAMPAIGNS WITH PANDAS

Wrap-up

ANALYZING MARKETING CAMPAIGNS WITH PANDAS

Jill Rosok
Data Scientist

Dataset

```
marketing = pd.read_csv('marketing.csv')
print(marketing.head())
```

```
      user_id date_served    channel          variant  conv  \
0  a100000029  2018-01-01  House Ads  personalization   True
1  a100000030  2018-01-01  House Ads  personalization   True
2  a100000031  2018-01-01  House Ads  personalization   True
3  a100000032  2018-01-01  House Ads  personalization   True
4  a100000033  2018-01-01  House Ads  personalization   True

  language_displayed preferred_language    age_group
0            English            English   0-18 years
1            English            English  19-24 years
2            English            English  24-30 years
3            English            English  30-36 years
4            English            English  36-45
```

Preprocessing

- Feature engineering
- Resolving errors in the data

Marketing metrics

$$\text{Conversion rate} = \frac{\text{Number of people who convert}}{\text{Total number of people who we market to}}$$

$$\text{Retention rate} = \frac{\text{Number of people who remain subscribed}}{\text{Total number of people who converted}}$$

Customer segmentation

```
marketing.groupby(['channel', 'age_group'])\
                                ['user_id'].count()
```

Dip in conversion rate?

```
house_ads = marketing[marketing['channel'] == 'House Ads']

language = conversion_rate(house_ads,
                           ['date_served',
                            'language_displayed'])
```

You analyzed an A/B test

- Lift
- T-tests

Good luck!

ANALYZING MARKETING CAMPAIGNS WITH PANDAS