

AB testing at Vungle

Write team members who contributed (Submit one copy per team):

Bhoomi Shah

Akshaya Ramprasad

Nivedya Pillai

Vaishnavi Sawant

Ardha Anil

1. What are the parties participated in the in-app mobile ad market? How does the in-app mobile work?

Parties participating in the in-app mobile ad market are:

- The user of the mobile device (**user**)
- The owner of the app being used (**publisher**)
- The sponsor of the video ad the user was exposed to (**advertiser**)
- The platform that matched the choice of ad to a specific user (e.g., **Vungle**)

The in-app mobile advertising market works in the following way:

1. User opens an app(owned by a publisher of that app)
2. The app sends a request to ad-serving platform (eg: Vungle) for an ad to show on the screen
3. Vungle would determine which ad to send back to the app based on the request it received. To match an ideal ad, it looks for the type of app through which request was sent(for eg: gaming, food delivery, etc.), the user(location, preferences, past behavior, etc) and the list of advertisers. It picks the best ad via an algorithm to get an install or a click from the user. The selected ad is sent back to the app.
4. App displays the ad received (assuming the screen on the device is still the one when request was sent). If atleast 80% of ad was watched, it leads to increase in completion rate.
5. If the user clicks (leads to increase in CTR) and then also installs(leads to increase in conversion rate), the ad payment workflow kicks in.
6. The revenues generated from the ad campaign are shared between the publisher of the app and Vungle(and are paid by advertiser). Publishers typically receive a percentage of the advertising revenue, while the rest goes to the Vungle.

2. How does Vungle make money? What is the typical measure of the effectiveness of an app-promotion and the success of the serving platform?

Vungle makes money through ads at all different points along the ad-serving journey. These are:

1. CPI (cost per install)
2. CPC (cost per click)
3. CPCV (cost per completed view)
4. CPM (cost per 1,000 views)

The typical measures for effectiveness of an app-promotion and the success of the serving platform are:

1. Fill Rate - Percentage of Ad Requests that are fulfilled (Successful response from the ad server)
2. Completion Rate - Percentage of users who watch a significant portion of the video ad ($\geq 80\%$)
3. Click-through Rate - Percentage of users who clicked the ad after viewing it
4. Conversion Rate: Percentage of users who not only clicked on an ad but also took the desired action, such as installing the promoted app.
5. eRPM (Effective Revenue per 1,000 Impressions): Combined revenue earned by both the publisher and Vungle (the ad platform) for every 1,000 ad impressions.

3. How does the new algorithm differ from the existing algorithm?

The new ad-serving algorithm (B) built by Kritzer and Guerin used a data science approach that would use historical information about users, publishers, and install rates to determine which ad campaign to serve in order to increase the chance of a conversion and, more specifically, eRPM. If proved successful, implementing it would require regular updates to the model (something like a feedback learning model) by a data scientist, most likely Guerin himself. As for the Vungle's existing algorithm (A), not much has been mentioned but it does not use a data science approach. Thus, we can assume it used a fixed set of rules to decide which ad to serve.

4. What are the key questions Vungle would like to know through the experiment?

Vungle's A/B testing experiment tries to answer several key questions that are crucial for looking at the performance of the new ad-serving algorithm (algorithm B) compared to the existing algorithm (algorithm A).

1. Does Algorithm B generate more revenue than Algorithm A? Vungle wants to see whether algorithm B leads to higher eRPM (effective revenue per 1,000 impressions) in comparison

to algorithm A. This question assesses whether the new algorithm can improve the financial performance of ad campaigns. Infact, in just two weeks, B was looking pretty good. Its daily eRPM was on average \$0.134 higher than algorithm A.

2. Does Algorithm B Improve Click-Through Rates (CTR)?: Vungle also wants to understand if algorithm B can achieve a higher click-through rate, indicating that it can drive more user engagement and interactions with ads.
3. Is Algorithm B Better at Converting Impressions into Installs?: The experiment seeks to determine whether algorithm B can increase the conversion rate, which represents the percentage of users who install advertised apps after interacting with the ads.
4. What Is the Impact on Fill Rate?: Vungle wants to understand if algorithm B affects the fill rate, which measures the percentage of ad requests that are successfully filled with ads. A higher fill rate is important for maximizing ad inventory utilization. If it is lower, it would mean the algorithm might be effective in eRPM and CTR but is lacking in performance/needs more computing to serve more requests.

5. Import the data provided 'Vungle.xlsx' and display the first 5 observations. Then, calculate completion rate, click-through rate (CTR), and conversion rate.

- Hint: `data['var1'] = data.var2/data.var3`

```
In [ ]: import pandas as pd
import numpy as np
from scipy import stats
```

```
In [ ]: df = pd.read_excel('Vungle.xlsx')
df.head()
```

```
Out[ ]:
```

	date	strategy	impression	complete	click	install	erpm
0	2014-06-01	A	6777407	5978434	345309	31119	3.327
1	2014-06-02	A	6004310	5331727	299732	24601	2.943
2	2014-06-03	A	5832627	5193549	291384	24220	3.025
3	2014-06-04	A	5875702	5227917	295099	23382	2.985
4	2014-06-05	A	6843405	6111378	339529	27725	3.076

```
In [ ]: #Completion rate
completion_rate = (df['complete'] / df['impression']).mean()
print(f"Completion Rate: {completion_rate:.2%}")
```

Completion Rate: 89.03%

```
In [ ]: #Click-through rate (CTR)
ctr = (df['click'] / df['impression']).mean()
print(f"Click-Through Rate (CTR): {ctr:.2%}")
```

Click-Through Rate (CTR): 5.00%

```
In [ ]: #Conversion rate
conversion_rate = (df['install'] / df['impression']).mean()
print(f"Conversion Rate: {conversion_rate:.2%}")
```

Conversion Rate: 0.38%

6. What are the average eRPM, completion, conversion, and CTR for conditions A and B?

- Hint: `data.groupby('var1')['var2'].mean()`

```
In [ ]: avg_erpm = df.groupby("strategy")["erpm"].mean()
avg_erpm
```

```
Out[ ]: strategy
A      3.3471
B      3.4590
Name: erpm, dtype: float64
```

```
In [ ]: # Filter data for strategy A
strategy_a_data = df[df['strategy'] == 'A']

# Filter data for strategy B
strategy_b_data = df[df['strategy'] == 'B']
```

```
In [ ]: # Average completion rate for strategy A and B
average_completion_rate_strategy_a = (strategy_a_data['complete'] / strategy_a_data['i
average_completion_rate_strategy_b = (strategy_b_data['complete'] / strategy_b_data['i

print(f"Average Completion Rate for strategy A: {average_completion_rate_strategy_a:.2
print(f"Average Completion Rate for strategy B: {average_completion_rate_strategy_b:.2

# Average CTR for strategy A and B
average_ctr_strategy_a = (strategy_a_data['click'] / strategy_a_data['impression']).me
average_ctr_strategy_b = (strategy_b_data['click'] / strategy_b_data['impression']).me

print(f"Average CTR for strategy A: {average_ctr_strategy_a:.2%}")
print(f"Average CTR for strategy B: {average_ctr_strategy_b:.2%}")

# Average conversion rate for strategy A and B
average_conversion_rate_strategy_a = (strategy_a_data['install'] / strategy_a_data['in
average_conversion_rate_strategy_b = (strategy_b_data['install'] / strategy_b_data['in

print(f"Average Conversion Rate for strategy A: {average_conversion_rate_strategy_a:.2
print(f"Average Conversion Rate for strategy B: {average_conversion_rate_strategy_b:.2
```

Average Completion Rate for strategy A: 89.27%

Average Completion Rate for strategy B: 88.78%

Average CTR for strategy A: 5.07%

Average CTR for strategy B: 4.92%

Average Conversion Rate for strategy A: 0.40%

Average Conversion Rate for strategy B: 0.35%

7. Test whether mean eRPM is equal to 3.4 for condition A and B. State the null and alternative hypotheses, and your conclusion with supporting reason. (Use 5% significance level)

- Hint: Subset (divide) the data into two
 - `dataA=data[data.strategy=="A"]`
 - `dataB=data[data.strategy=="B"]`

```
In [ ]: stats.ttest_1samp(strategy_a_data['erpm'], 3.4)
```

```
Out[ ]: TtestResult(statistic=-1.3378706323475629, pvalue=0.19133605188963274, df=29)
```

```
In [ ]: stats.ttest_1samp(strategy_b_data['erpm'], 3.4)
```

```
Out[ ]: TtestResult(statistic=0.9382534519621959, pvalue=0.35586335556174653, df=29)
```

- H0: Mean eRPM is equal to 3.4 for condition A and B.
- HA: Mean eRPM is not equal to 3.4 for condition A and B.
- Conclusion: We **do not reject** H0 because p-value for both conditions > 5%. The mean eRPM for Condition A and B are equal to 3.4

8. Test whether mean eRPMs are different for condition A and B. State the null and alternative hypotheses, and your conclusion with supporting reason. (Use 5% significance level)

```
In [ ]: stats.ttest_ind(strategy_a_data["erpm"], strategy_b_data["erpm"]) # t-statistic and p
```

```
Out[ ]: TtestResult(statistic=-1.5064382333172264, pvalue=0.13738210974199117, df=58.0)
```

- H0: Mean eRPM for condition A and B are same.
- HA: Mean eRPM for condition A and B are different.
- Conclusion: We **do not reject** H0 because p-value > 5%. The mean eRPM for condition A and B are not different

9. Test whether mean conversion rates are different for condition A and B. State the null and alternative hypotheses, and your conclusion with supporting reason. (Use 5% significance level)

```
In [ ]: strategy_a_data["conversion_rates"] = strategy_a_data["install"]/strategy_a_data["impr"]
strategy_b_data["conversion_rates"] = strategy_b_data["install"]/strategy_b_data["impr"]
stats.ttest_ind(strategy_a_data["conversion_rates"], strategy_b_data["conversion_rates"])
```

```
Out[ ]: TtestResult(statistic=8.81563097620209, pvalue=2.6750870126681745e-12, df=58.0)
```

- H0: Mean conversion rates for condition A and B are same.
- HA: Mean conversion rates for condition A and B are different.
- Conclusion: We **reject** H0 because p-value is extremely small(2.6×10^{-12}). Mean conversion rates for condition A and B are different.

10. What would you advise Jaffer regarding the performance of the new data science algorithm?

Our conclusion and advice to Jaffer based on the above tests would be:

eRPM:

As mentioned in the article, the daily eRPM in the first 2 weeks was on average \$0.134 higher than algorithm A.

But, the statistical test above for mean eRPM shows that both conditions A and B have the same eRPM(3.4). This suggests that, on average, both strategies perform similarly in terms of generating revenue. Therefore, the choice between the two strategies should not be based solely on eRPM but should consider other factors, such as user experience and ad effectiveness.

The two-sample independent test between A and B also says eRPM are not different. This means that, from a revenue perspective, both strategies are equally effective.

Conversion Rate:

There is a statistically significant difference(the 9th question rejected the null hypothesis) in the mean conversion rates between both conditions. Thus, the mean conversion rate of condition A(0.4%) is statistically higher than B(0.3%)

Final Advice:

- Continue to monitor and gather user feedback and engagement metrics to assess how users perceive ads served by each strategy.
- Evaluate the scalability and sustainability of strategy B to ensure it can adapt to changing market dynamics and user behaviors over time
- Not to completely switch to algorithm B right away but establish a feedback loop for continuous improvement, allowing for regular updates and fine-tuning of strategy B based on real-world data and performance.