# **AB** Testing

#### Question 01

We are running an experiment at an item-level, which means all users who visit will see the same page, but the layout of different item pages may differ. Compare this table to the assignment events we captured for user\_level\_testing. Does this table have everything you need to compute metrics like 30-day view-binary?

```
SELECT * FROM dsv1069.final_assignments_qa
```

**ANSWER:** No, the record creation date is required.

#### Question 02

Reformat the final\_assignments\_qa to look like the final\_assignments table, filling in any missing values with a placeholder of the appropriate data type.

```
SQL CODE
SELECT item_id,
   test a AS test assignment,
   'test a' AS test number,
   CAST('2022-11-15 00:00:00' AS timestamp) AS dummy test start date
FROM dsv1069.final assignments qa
UNION ALL
SELECT item id,
   test_b AS test_assignment,
   'test b' AS test number,
   CAST('2022-11-15 00:00:00' AS timestamp) AS dummy test start date
FROM dsv1069.final assignments ga
UNION ALL
SELECT item id,
   test_c AS test_assignment,
   'test_c' AS test_number,
   CAST('2022-11-15 00:00:00' AS timestamp) AS dummy test start date
FROM dsv1069.final_assignments_qa
UNION ALL
SELECT item id,
   test_d AS test_assignment,
   'test d' AS test number,
   CAST('2022-11-15 00:00:00' AS timestamp) AS dummy test start date
FROM dsv1069.final assignments qa
UNION ALL
SELECT item_id,
   test_e AS test_assignment,
   'test e' AS test number,
   CAST('2022-11-15 00:00:00' AS timestamp) AS dummy test start date
FROM dsv1069.final assignments qa
UNION ALL
SELECT item_id,
   test_f AS test_assignment,
   'test f' AS test number,
   CAST('2022-11-15 00:00:00' AS timestamp) AS dummy_test_start_date
FROM dsv1069.final_assignments_qa;
```

#### Question 03

Use the final\_assignments table to calculate the order binary for the 30 day window after the test assignment for item\_test\_2 (You may include the day the test started)

```
SQL CODE
SELECT order_binary.test_assignment,
    COUNT(DISTINCT order binary.item id) AS num orders,
    SUM(order_binary.ob30d) AS sum_orders_bin_30d
FROM
 SELECT assignments.item id,
     assignments.test_assignment,
     MAX(CASE
         WHEN (DATE(orders.created at)-DATE(assignments.test start date)) BETWEEN 1 AND 30
THEN 1
         ELSE 0
       END) AS ob30d
 FROM dsv1069.final_assignments AS assignments
 LEFT JOIN dsv1069.orders AS orders
  ON assignments.item_id=orders.item_id
 WHERE assignments.test number='item test 2'
 GROUP BY assignments.item id,
      assignments.test_assignment) AS order_binary
GROUP BY order_binary.test_assignment
```

		RESULT	
test_assignment num_orders sum_orders_bin_30d	test_assignment	num_orders	sum_orders_bin_30d
0 1130 331	0	1130	331
1 1068 306	1	1068	306

#### Question 04

Use the final\_assignments table to calculate the view binary, and average views for the 30 day window after the test assignment for item\_test\_2. (You may include the day the test started).

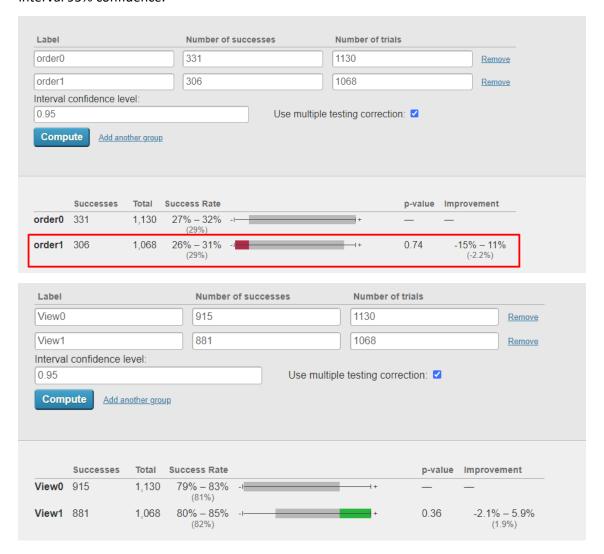
```
SQL CODE
SELECT view binary.test assignment,
   COUNT(DISTINCT view binary.item id) AS num views,
   SUM(view binary.view bin 30d) AS sum view bin 30d,
   AVG(view_binary.view_bin_30d) AS avg_view_bin_30d
FROM
 (SELECT assignments.item id,
     assignments.test_assignment,
     MAX(CASE
         WHEN (DATE(views.event time)-DATE(assignments.test start date)) BETWEEN 1 AND 30
THEN 1
         ELSE 0
       END) AS view bin 30d
 FROM dsv1069.final_assignments AS assignments
 LEFT JOIN dsv1069.view_item_events AS views
  ON assignments.item_id=views.item_id
 WHERE assignments.test_number='item_test_2'
 GROUP BY assignments.item_id,
```

assignments.test\_assignment
ORDER BY item\_id) AS view\_binary
GROUP BY view\_binary.test\_assignment

RESULT				
test_assignment	num_views	sum_view_bin_30d	avg_view_bin_30d	
0	1130	915	0.8097	
1	1068	881	0.8249	

### Question 05

Use the https://thumbtack.github.io/abba/demo/abba.html to compute the lifts in metrics and the p-values for the binary metrics ( 30 day order binary and 30 day view binary) using a interval 95% confidence.



**ANSWER:** 

- For orders\_bin: lift is -15% 11% (-2.2%) and pval is 0.74
- $\bullet$  For views\_bin: lift is and pval is -2.1% 5.9% (1.9%) and pval is 0.36

• There is no significant difference for the test item\_test\_2 in the binary metrics of orders and views.

## Question 06

