Assignment Tasks

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. Please follow the steps below and good luck!

1. Compare the final_assignments_qa table to the assignment events we captured for user_level_testing. Write an answer to the following question: Does this table have everything you need to compute metrics like 30-day view-binary?

<u>Answer:</u> No, it does not.

2. Write a query and table creation statement to make final_assignments_qa look like the final_assignments table. If you discovered something missing in part 1, you may fill in the value with a place holder of the appropriate data type.

```
SELECT
  item_id,
  test_a AS test_assignment,
  (CASE WHEN test_a IS NOT NULL then 'test_a'
    ELSE NULL
  END) AS test_number,
  (CASE WHEN test_a IS NOT NULL then '2013-01-05 00:00:00'
    ELSE NULL
  END) AS test_start_date
FROM
  dsv1069.final_assignments_qa
UNION
  SELECT
    item_id,
    test_b AS test_assignment,
    (CASE WHEN test_b IS NOT NULL then 'test_b'
      ELSE NULL
    END) AS test_number,
    (CASE WHEN test_b IS NOT NULL then '2013-01-05 00:00:00'
      ELSE NULL
    END) AS test_start_date
  FROM
    dsv1069.final_assignments_qa
  UNION
    SELECT
      item_id,
      test_c AS test_assignment,
      (CASE WHEN test_c IS NOT NULL then 'test_c'
        ELSE NULL
      END) AS test_number,
  (CASE WHEN test_c IS NOT NULL then '2013-01-05 00:00:00'
    ELSE NULL
  END) AS test_start_date
FROM
  dsv1069.final_assignments_qa
UNION
  SELECT
    item_id,
    test_d AS test_assignment,
    (CASE WHEN test_d IS NOT NULL then 'test_d'
      ELSE NULL
    END) AS test_number,
    (CASE WHEN test_d IS NOT NULL then '2013-01-05 00:00:00'
      ELSE NULL
    END) AS test_start_date
  FROM
    dsv1069.final_assignments_qa
  UNION
  SELECT
    item_id,
    test_e AS test_assignment,
    (CASE WHEN test_e IS NOT NULL then 'test_e'
      ELSE NULL
    END) AS test_number,
    (CASE WHEN test_e IS NOT NULL then '2013-01-05 00:00:00'
      ELSE NULL
    END) AS test_start_date
  FROM
    dsv1069.final_assignments_qa
  UNION
    SELECT
      item_id,
      test_f AS test_assignment,
      (CASE WHEN test_f IS NOT NULL then 'test_f'
        ELSE NULL
      END) AS test_number,
      (CASE WHEN test_f IS NOT NULL then '2013-01-05 00:00:00'
        ELSE NULL
      END) AS test_start_date
    FROM
        dsv1069.final_assignments_qa;
```

3. 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)

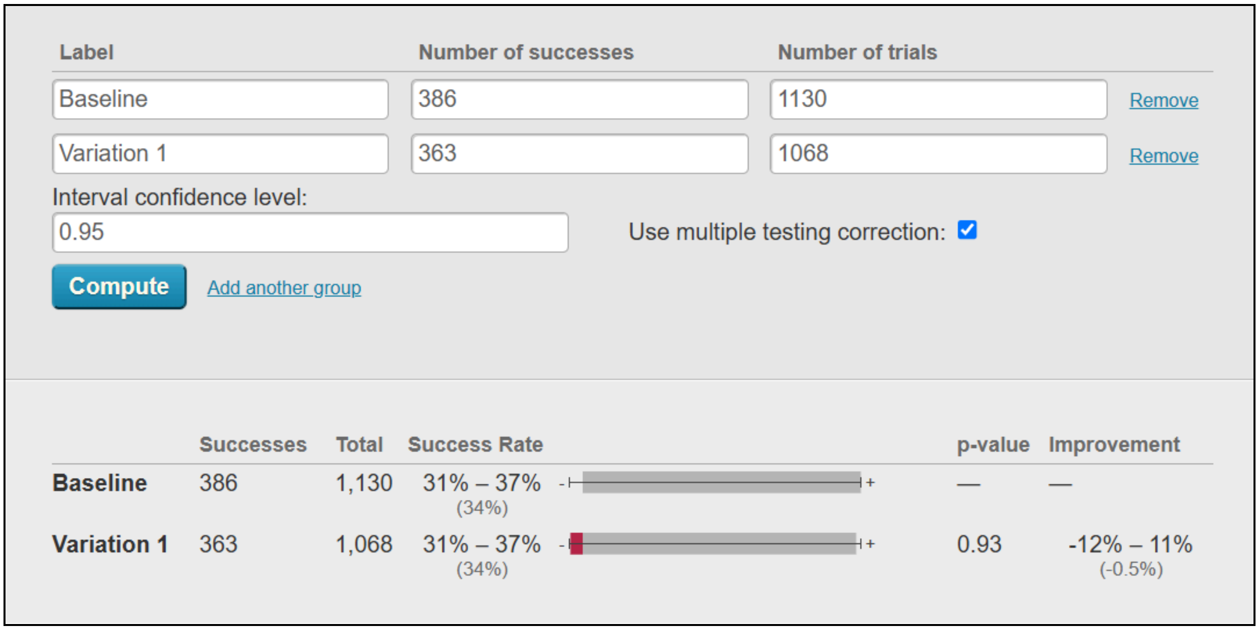
```
SELECT
  test_assignment,
  COUNT(DISTINCT item_id) AS number_items,
  SUM(order_binary) AS orders_30days
FROM
  (
  SELECT
    item_test_2.item_id,
    item_test_2.test_assignment,
    item_test_2.test_number,
    item_test_2.test_start_date,
    item_test_2.created_at,
    MAX(CASE WHEN(created_at > test_start_date
      AND DATE_PART('day', created_at - test_start_date) <= 30)
      THEN 1 ELSE 0
      END) AS order_binary
  FROM
    (SELECT
      fs.*,
      DATE(ord.created_at) AS created_at
    FROM
      dsv1069.final_assignments AS fs
    LEFT JOIN
      dsv1069.orders AS ord
    ON
      fs.item_id = ord.item_id
    WHERE
      test_number = 'item_test_2') AS item_test_2
GROUP BY
  item_test_2.item_id,
  item_test_2.test_assignment,
  item_test_2.test_number,
  item_test_2.test_start_date,
  item_test_2.created_at) AS order_binary
GROUP BY
  test_assignment;
```

4. 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)

```
SELECT
  item_test_2.item_id,
  item_test_2.test_assignment,
  item_test_2.test_number,
  MAX(CASE WHEN (view_date > test_start_date
   AND DATE_PART('day', view_date - test_start_date) <= 30)
   THEN 1 ELSE 0
   END) AS view_binary
 FROM
  (SELECT
   fs.*,
   DATE(events.event_time) AS view_date
  FROM
   dsv1069.final_assignments AS fs
  LEFT JOIN
   (SELECT
    event_time,
   CASE WHEN parameter_name = 'item_id'
    THEN CAST(parameter_value AS NUMERIC)
    ELSE NULL
   END AS item_id
   FROM
    dsv1069.events
   WHERE
    event_name = 'view_item') AS events
  ON
   fs.item_id = events.item_id
  WHERE
   test_number = 'item_test_2') AS item_test_2
 GROUP BY
  item_test_2.item_id,
  item_test_2.test_assignment,
  item_test_2.test_number) AS view_binary
```

5. 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.

30 day order binary:



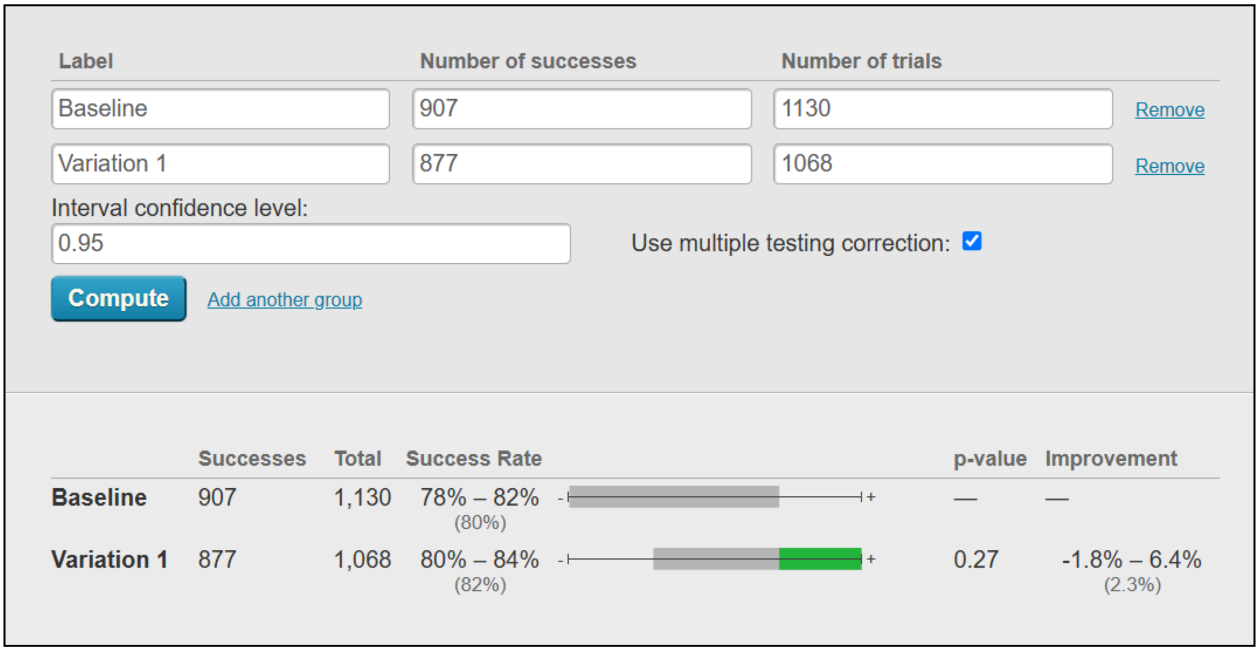
Label	Number of successes	Number of trials	
Baseline	386	1130	Remove
Variation 1	363	1068	Remove

Interval confidence level: 0.95

Use multiple testing correction: ☑

	Successes	Total	Success Rate	p-value	Improvement
Baseline	386	1,130	31% – 37% (34%)	—	—
Variation 1	363	1,068	31% – 37% (34%)	0.93	-12% – 11% (-0.5%)

30 day view binary:



Label	Number of successes	Number of trials	
Baseline	907	1130	Remove
Variation 1	877	1068	Remove

Interval confidence level: 0.95

Use multiple testing correction: ☑

	Successes	Total	Success Rate	p-value	Improvement
Baseline	907	1,130	78% – 82% (80%)	—	—
Variation 1	877	1,068	80% – 84% (82%)	0.27	-1.8% – 6.4% (2.3%)