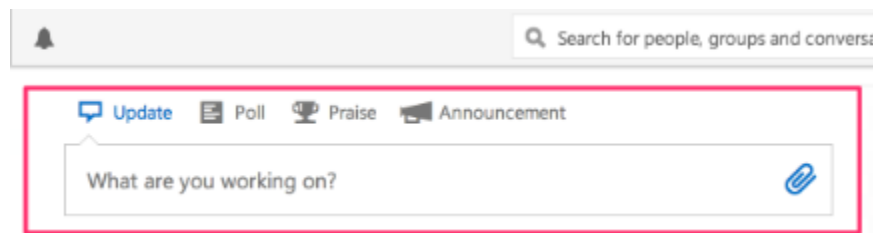


Project Description: [Validating A/B Test Results | SQL Analytics Training - Mode](#)

The primary job of A/B testing in a business context is to make predictions. A/B tests help analysts and product managers better understand a feature's effect on user behavior and overall user experience. In this case, Yammer revised its feature where users post messages (see picture). To test it, the company ran a test from June 1 - June 30 and selected users to be part of its experiment.  $\frac{2}{3}$  of the population sample is part of the control group and did not see changes to the service.  $\frac{1}{3}$  of the population is part of the treatment group and saw the new features. According to the data, the treatment group posted 50% more often than the control group. Yammer wants to verify that its results are not too good to be true.



Question 1: How did the changes affect over metrics between the control group and treatment group?

A test script was provided to perform all of the statistical calculations. The additionally provided data also shows that users in the control group created 2.7 posts/day, while individuals in the test group wrote 4 posts/day. This is where the 50% increase comes from. All other data in the table shows the test is statistically valid.

```
SELECT c.experiment,
       c.experiment_group,
       c.users,
       c.total_treated_users,
       ROUND(c.users/c.total_treated_users,4) AS treatment_percent,
       c.total,
       ROUND(c.average,4)::FLOAT AS average,
       ROUND(c.average - c.control_average,4) AS rate_difference,
       ROUND((c.average - c.control_average)/c.control_average,4) AS rate_lift,
       ROUND(c.stdev,4) AS stdev,
       ROUND((c.average - c.control_average) /
             SQRT((c.variance/c.users) + (c.control_variance/c.control_users))
             ,4) AS t_stat,
       (1 - COALESCE(nd.value,1))*2 AS p_value
FROM (
SELECT *,
       MAX(CASE WHEN b.experiment_group = 'control_group' THEN b.users ELSE NULL END)
OVER () AS control_users,
       MAX(CASE WHEN b.experiment_group = 'control_group' THEN b.average ELSE NULL
END) OVER () AS control_average,
       MAX(CASE WHEN b.experiment_group = 'control_group' THEN b.total ELSE NULL END)
OVER () AS control_total,
```

```

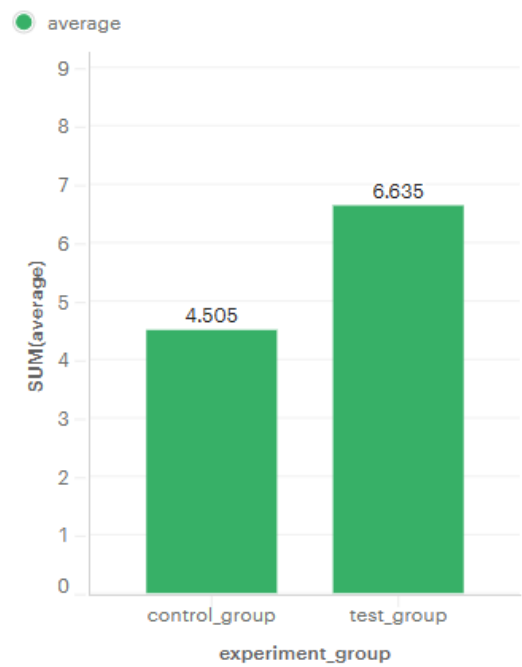
        MAX(CASE WHEN b.experiment_group = 'control_group' THEN b.variance ELSE NULL
END) OVER () AS control_variance,
        MAX(CASE WHEN b.experiment_group = 'control_group' THEN b.stdev ELSE NULL END)
OVER () AS control_stdev,
        SUM(b.users) OVER () AS total_treated_users
    FROM (
SELECT a.experiment,
       a.experiment_group,
       COUNT(a.user_id) AS users,
       AVG(a.metric) AS average,
       SUM(a.metric) AS total,
       STDDEV(a.metric) AS stdev,
       VARIANCE(a.metric) AS variance
    FROM (
SELECT ex.experiment,
       ex.experiment_group,
       ex.occurred_at AS treatment_start,
       u.user_id,
       u.activated_at,
       COUNT(CASE WHEN e.event_name = 'send_message' THEN e.user_id ELSE NULL END) AS
metric
    FROM (SELECT user_id,
                 experiment,
                 experiment_group,
                 occurred_at
          FROM tutorial.yammer_experiments
          WHERE experiment = 'publisher_update'
        ) ex
    JOIN tutorial.yammer_users u
      ON u.user_id = ex.user_id
    JOIN tutorial.yammer_events e
      ON e.user_id = ex.user_id
     AND e.occurred_at >= ex.occurred_at
     AND e.occurred_at < '2014-07-01'
     AND e.event_type = 'engagement'
    GROUP BY 1,2,3,4,5
        ) a
    GROUP BY 1,2
        ) b
        ) c
    LEFT JOIN benn.normal_distribution nd
      ON nd.score = ABS(ROUND((c.average - c.control_average)/SQRT((c.variance/c.users)
+ (c.control_variance/c.control_users)),3))

```

I am able to change which feature I'm measuring between the two groups by accessing the subquery. In this case, I wanted to see, on average, how many messages the users liked daily. The test group is still more active than the control group.

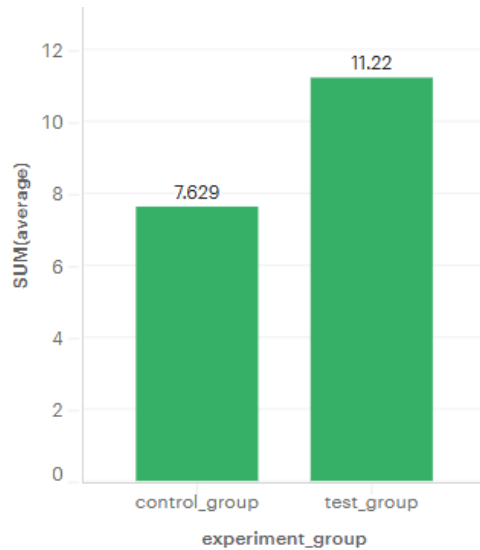


Output when I measured, on average, how many times users check their inbox.



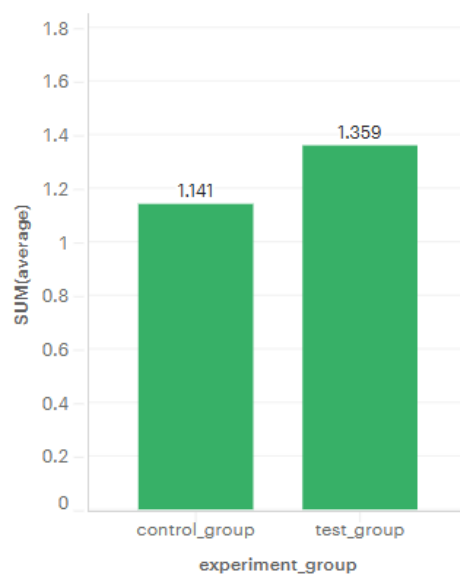
average	rate_difference	rate_lift	stdev	t_stat	p_value
4.5052	0	0	5.8563	0	1
6.6349	2.1297	0.4727	7.4326	7.3173	0

Output when I measured, on average, how many times users visit their homepage.



average	rate_difference	rate_lift	stdev	t_stat	p_value
7.6294	0	0	9.4019	0	1
11.2179	3.5885	0.4703	12.2647	7.5184	0

Output when I measured, on average, how many search queries are written daily.



average	rate_difference	rate_lift	stdev	t_stat	p_value
1.1409	0	0	2.7308	0	1
1.3592	0.2184	0.1914	3.0848	1.755	0.07926

If the changes to the posting feature caused all of these other metrics in the testing group to skyrocket, then Yammer should absolutely roll it out. That is probably too good to be true though, and it's likely that the experimental group is simply very active.

## Question 2: Did the study compare individual users the same way?

Studies sometimes clump newer and older users together. They also assess laptop and mobile users the same way, which is a problem because Yammer's service might not be as mobile friendly. Thankfully, the percentage of users represented is consistent across both test groups.

```
SELECT experiment_group,
  100 * COUNT(DISTINCT CASE WHEN device IN ('macbook pro','lenovo
thinkpad','macbook air','dell inspiron notebook',
      'asus chromebook','dell inspiron desktop','acer aspire
notebook','hp pavilion desktop','acer aspire desktop','mac mini')
      THEN user_id ELSE NULL END)/COUNT(*) AS computer_percentage,
  100 * COUNT(DISTINCT CASE WHEN device IN ('iphone 5','samsung galaxy
s4','nexus 5','iphone 5s','iphone 4s','nokia lumia 635',
      'htc one','samsung galaxy note','amazon fire phone') THEN user_id
      ELSE NULL END)/COUNT(*) AS phone_percentage,
  100 * COUNT(DISTINCT CASE WHEN device IN ('ipad air','nexus 7','ipad
mini','nexus 10','kindle fire','windows surface',
      'samsung galaxy tablet') THEN user_id ELSE NULL END)/COUNT(*) AS
tablet_percentage
FROM tutorial.yammer_experiments
GROUP BY 1
```

	experiment_group	computer_percentage	phone_percentage	tablet_percentage
1	control_group	57	29	12
2	test_group	55	33	10

My second approach is to break down the percentages of users introduced into the experiment one day after account creation, one week, one month, etc.

## Treatment Group

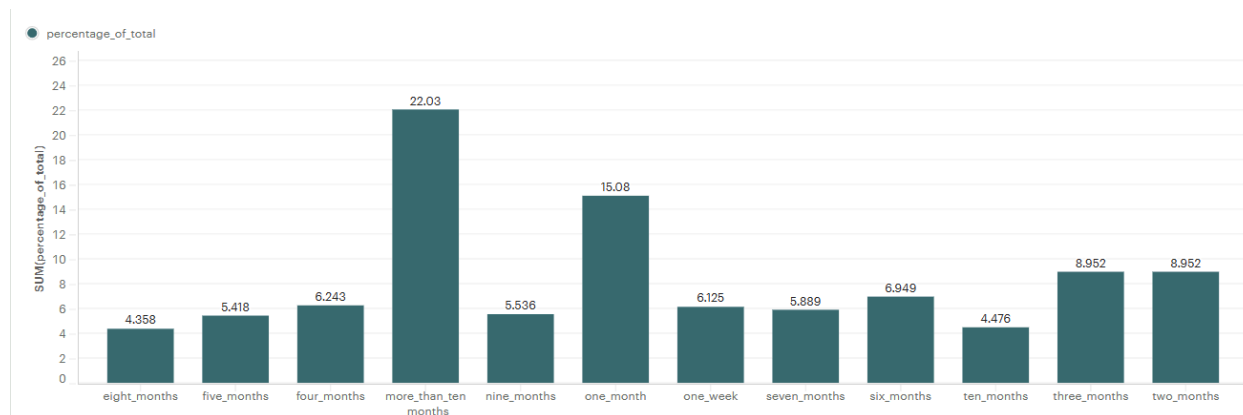
```
SELECT sub.account_age,
COUNT (sub.account_age) AS num_participants
FROM (
  SELECT CASE WHEN e.occurred_at < u.created_at + INTERVAL '1 day' THEN
'one_day'
      WHEN e.occurred_at < u.created_at + INTERVAL '1 week' THEN 'one_week'
```

```

        WHEN e.occurred_at < u.created_at + INTERVAL '1 month' THEN
'one_month'
        WHEN e.occurred_at < u.created_at + INTERVAL '2 months' THEN
'two_months'
        WHEN e.occurred_at < u.created_at + INTERVAL '3 months' THEN
'three_months'
        WHEN e.occurred_at < u.created_at + INTERVAL '4 months' THEN
'four_months'
        WHEN e.occurred_at < u.created_at + INTERVAL '5 months' THEN
'five_months'
        WHEN e.occurred_at < u.created_at + INTERVAL '6 months' THEN
'six_months'
        WHEN e.occurred_at < u.created_at + INTERVAL '7 months' THEN
'seven_months'
        WHEN e.occurred_at < u.created_at + INTERVAL '8 months' THEN
'eight_months'
        WHEN e.occurred_at < u.created_at + INTERVAL '9 months' THEN
'nine_months'
        WHEN e.occurred_at < u.created_at + INTERVAL '10 months' THEN
'ten_months'
        ELSE 'more_than_ten_months' END AS account_age
FROM tutorial.yammer_users u
JOIN tutorial.yammer_experiments e
  ON u.user_id = e.user_id
  AND e.experiment_group = 'test_group'
) sub
GROUP BY 1
ORDER BY 1

```

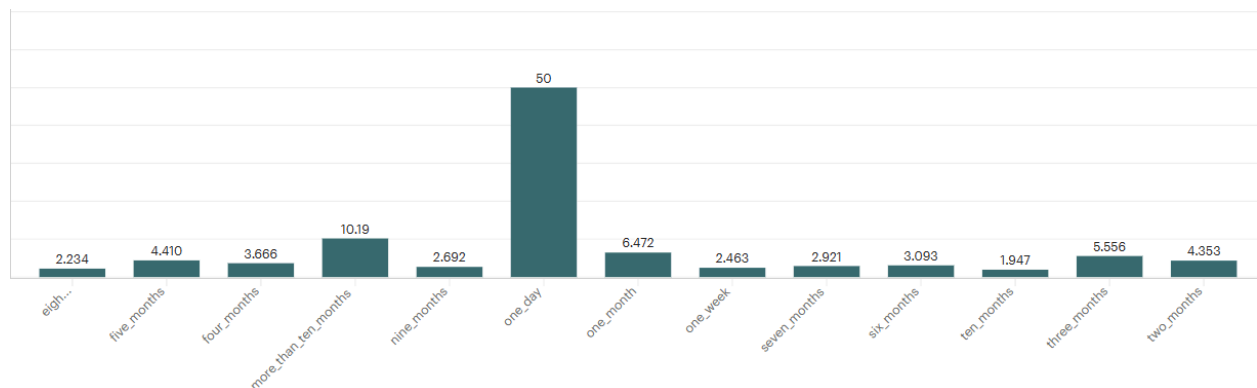
	account_age	num_participants	percentage_of_total
1	eight_months	37	4.3581
2	five_months	46	5.4181
3	four_months	53	6.2426
4	more_than_ten_mont...	187	22.0259
5	nine_months	47	5.5359
6	one_month	128	15.0766
7	one_week	52	6.1249
8	seven_months	50	5.8893
9	six_months	59	6.9494
10	ten_months	38	4.4759
11	three_months	76	8.9517
12	two_months	76	8.9517



If you're having trouble reading the graph, the data shows that Yammer has a mixture of new and more veteran users in the test group. This explains why the posting rate is higher than the control group.

### Control Group (same script with a modification)

	account_age	num_participants	percentage_of_total
1	eight_months	39	2.2337
2	five_months	77	4.4101
3	four_months	64	3.6655
4	more_than_ten_mont...	178	10.1947
5	nine_months	47	2.6919
6	one_day	873	50
7	one_month	113	6.4719
8	one_week	43	2.4628
9	seven_months	51	2.9210
10	six_months	54	3.0928
11	ten_months	34	1.9473
12	three_months	97	5.5556
13	two_months	76	4.3528



Unlike the testing group, this population sample is skewed with over half of the users brought into the experiment just a day after creating their account. This obviously causes implications because they are still learning how to post, check their inbox, and access any other features in Yammer. While most of the data looks valid, the experiment should be redone by introducing more veteran accounts into the testing group.