Springboard SQL Assignment with Mode Analytics
Vicki M.
March 10, 2018

Case study: Best A/B Test

The problem and possible explanations:

This A/B test involves a website variation that results in a dramatically higher apparent engagement in messaging with the variation being tested versus the control group (1.4 times the frequency of the control, two-tailed t-stat. reported as 7.6245, $0.05 \leq p \leq .1$, df=1). The result is extraordinary, leading the company to wonder if there is an error involved. The new variation does not appear to lead to an increase in use, but the messaging rate is high from this group to a surprising degree for the number of users in comparison with the control.

A first hypothesis to explain these results would be that there is a bottleneck affecting users (since the summary statistics indicate the test variation has about half the number of users that the control does, and this is unexplained); are the users who surpass any technical hurdle in the process simply more avid and inclined to actively message?

A second hypothesis would be that perhaps some sort of glitch automatically sends out a message (in error? as an ad? etc.) by people who engaged with the variation. This seems much less likely, as it would require active development of such a thing and for no obvious reason.

A third hypothesis is that the test did not have users randomly allocated to each group. Obviously, the control group would not be influenced by a new condition, but the treatment group could work better for or be used more effectively by some users than for others. Since number of users is smaller, perhaps some devices, languages, or locations do not work well with it, and perhaps the devices or locations that do interact well with it are also ones that are easy to message with or consist of a subpopulation that is more inclined to send messages using this software. This is similar to the first hypothesis case but involves less of an effort component by users and passively selects more for a more communicative subpopulation.

The first hypothesis could be assessed by querying for message histories of users. Judging the second hypothesis would not be possible without more information, but it is a low-likelihood scenario. Querying for data on devices, locations, and languages could give insight into the third hypothesis. Of course, these hypotheses also depend on how one is viewing the available data; where it describes users as those to whom the version is shown, I am taking that to mean those who have logged in and seen it, because there is an imbalance between groups for what is described as "users", while the "total treatment group" numbers are even. This is not explained in the summary information. In any case, login history would matter.

A quick glance at users:

This query below shows that the actual allocation of users to control group was 1746 (as posted in the summary and versus 849 users described for the test condition), so my initial interpretation of what the “users” column meant was incorrect. It appears that the “users” data column refers to those samplings of users to whom the experiment was actually applied out of total populations of 2595 each, rather than those who accessed the site and “viewed” their group’s variant. Still unclear is why the users allocated to each are not balanced in number.

```
SELECT COUNT(occurred_at) AS count
FROM tutorial.yammer_experiments
WHERE experiment_group LIKE 'control_group'
```

Result is: count 1746

Since we know that "users" does not refer to what I had hypothesized in hypothesis one (something to do with users hitting a glitch when accessing content), the next thing to examine is whether there is a difference for each group in logging in and accessing content, before judging whether anything is odd about the rate of message sending. The following query tells us about the rate of login attempts and message sending between each group:

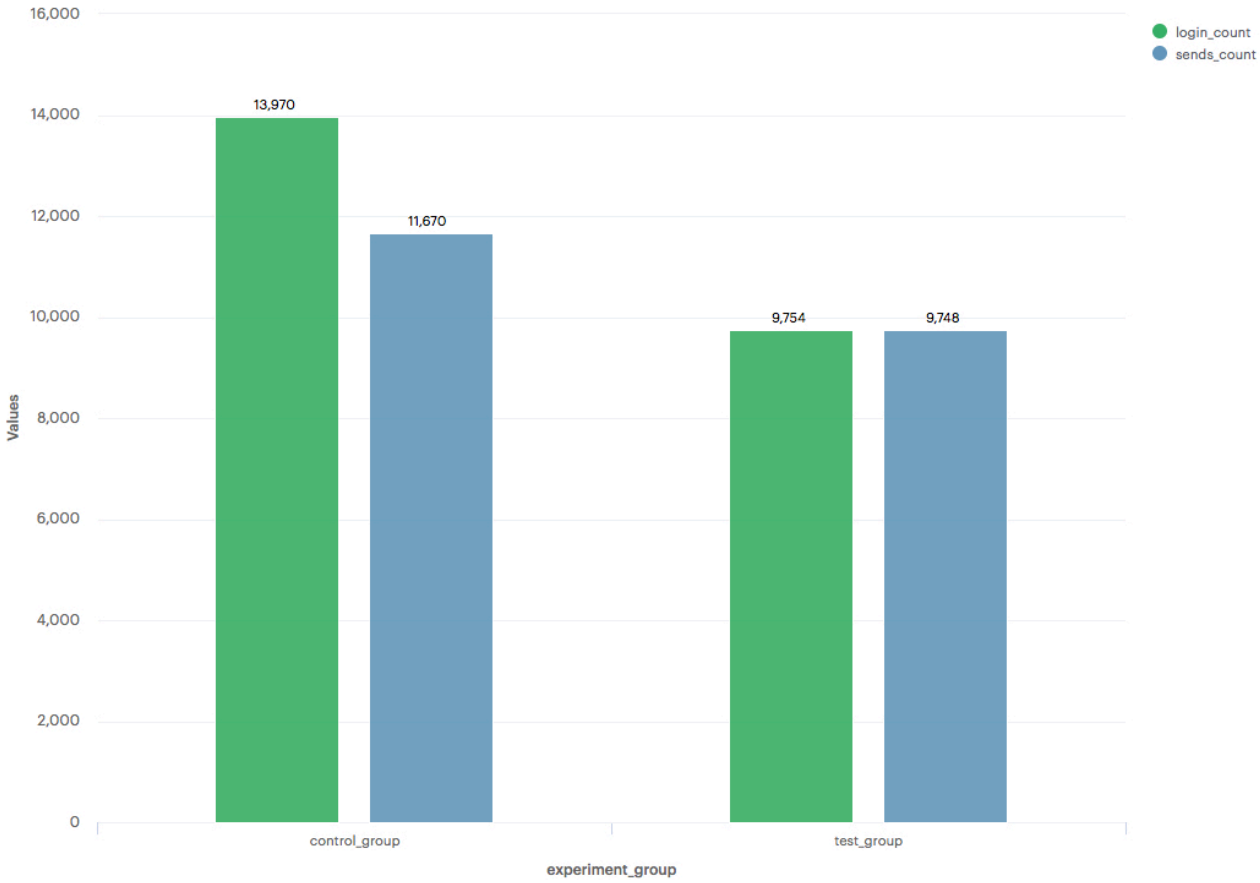
```
SELECT COUNT(CASE WHEN events.event_name = 'login' THEN 1 ELSE NULL END) AS login_count,
COUNT(CASE WHEN events.event_name = 'send_message' THEN 1 ELSE NULL END) AS sends_count,
experiments.experiment_group
FROM tutorial.yammer_events events
JOIN tutorial.yammer_experiments experiments
ON events.user_id = experiments.user_id
GROUP BY experiments.experiment_group
```

Results are:
control_group logins: 13970 messages sent: 11670
test_group logins: 9754 messages sent: 9748.

Login and message sending characteristics:

Logins and numbers of messages sent are roughly equal for the test group (but not exactly equal). The control group shows fewer messages sent than logins. Results for the test group make hypothesis two even less likely, because if a message was automatically sent from each (barring a few that failed to send), these results would also imply no one in the treatment sent an intentional message (which is possible if the service is not working, but there would likely be ample customer feedback in such a case). Hypothesis three, that there is a quality to the treatment group that makes it on the whole more likely to communicate than the control does, seems to be at play. This suggests a non-random group allocation. Figure 1 characterizes the results of this query.

Figure 1. Count of logins and sent messages by experiment group



A look at locations and device types:

The following query tells us how many locations and device types comprise each experimental group:

```
SELECT COUNT(DISTINCT events.location) AS locations,
COUNT(DISTINCT events.device) AS devices,
experiments.experiment_group
FROM tutorial.yammer_events events
JOIN tutorial.yammer_experiments experiments
ON events.user_id = experiments.user_id
GROUP BY experiments.experiment_group
```

Results are:
control_group locations: 47 device types: 26
test_group locations: 47 device types: 26.

Since these represent the totals possible for each of those categories, there is not a difference between the two groups in terms of which locations or devices are included in each. It would be possible for locations and devices to be distributed unevenly in each group, however.

The following query helps us assess if there is an obvious overrepresentation of any location in either group:

```
SELECT events.location,
   COUNT(*) AS count,
   experiments.experiment_group
 FROM tutorial.yammer_events events
 JOIN tutorial.yammer_experiments experiments
 ON events.user_id = experiments.user_id
 GROUP BY events.location, experiments.experiment_group
 ORDER BY events.location, experiments.experiment_group
```

The pivot table shown below allows us to view the allocation of members from each country in each location. There are more individuals from many locations who are in the control group, but that is expected since that group has so many more members than the test group does. That would not explain a higher frequency of communication from members represented in the test condition, nor really does the small occurrence of some higher numbers in some locations that are in the test group. A multivariate or other analysis would really tell us if there is a connection between location and experiment group. However, just by eye it is clear that no particular locations stand out as being associated with the test condition in such a way as to explain the substantially higher frequency of messaging by people in that group.

	experiment_group		
location	control_group	test_group	**Totals**
Argentina	896	248	**1144**
Australia	2164	2427	**4591**
Austria	1268	734	**2002**
Belgium	1162	620	**1782**
Brazil	3235	4020	**7255**
Canada	4075	1858	**5933**
Chile	231	306	**537**
Colombia	296	578	**874**
Denmark	531	180	**711**

	experiment_group		
location	control_group	test_group	**Totals**
Egypt	1187	541	**1728**
Finland	821	462	**1283**
France	6648	4496	**11144**
Germany	6895	6597	**13492**
Greece	552	55	**607**
Hong Kong	290	553	**843**
India	4036	1814	**5850**
Indonesia	2026	1815	**3841**
Iran	1637	682	**2319**
Iraq	107	717	**824**
Ireland	150	381	**531**
Israel	425	720	**1145**
Italy	5305	1886	**7191**
Japan	9899	6403	**16302**
Korea	2631	2328	**4959**
Malaysia	952	679	**1631**
Mexico	3004	2686	**5690**
Netherlands	1797	652	**2449**
Nigeria	419	469	**888**
Norway	815	482	**1297**
Pakistan	351	61	**412**
Philippines	515	188	**703**
Poland	1121	1567	**2688**
Portugal	440	355	**795**
Russia	3315	3951	**7266**
Saudi Arabia	1270	1441	**2711**
Singapore	666	28	**694**
South Africa	869	230	**1099**

	experiment_group		
location	control_group	test_group	**Totals**
Spain	1999	1633	**3632**
Sweden	1942	930	**2872**
Switzerland	1491	718	**2209**
Taiwan	1033	1531	**2564**
Thailand	461	708	**1169**
Turkey	527	403	**930**
United Arab Emirates	834	842	**1676**
United Kingdom	5236	5091	**10327**
United States	32674	25021	**57695**
Venezuela	529	578	**1107**
Totals	**118727**	**90665**	**20939**

If location does not appear to generate artifact in messaging frequency in either experimental condition, the next question is whether device type does. The following query helps us figure this out:

```
SELECT events.device,
   COUNT(*) AS count,
   experiments.experiment_group
 FROM tutorial.yammer_events events
 JOIN tutorial.yammer_experiments experiments
 ON events.user_id = experiments.user_id
 GROUP BY events.device, experiments.experiment_group
 ORDER BY events.device, experiments.experiment_group
```

As with location, device type does not clearly explain the higher messaging frequency shown by the test group, with users and device types presented in the pivot table below and organized by experiment group.

	experiment_group		
device	control_group	test_group	**Totals**
acer aspire desktop	1880	1413	**3293**
acer aspire notebook	3367	2335	**5702**

	experiment_group		
device	control_group	test_group	**Totals**
amazon fire phone	844	503	**1347**
asus chromebook	3459	2546	**6005**
dell inspiron desktop	4095	3081	**7176**
dell inspiron notebook	7525	6029	**13554**
hp pavilion desktop	3282	2387	**5669**
htc one	1747	995	**2742**
ipad air	2977	3144	**6121**
ipad mini	2143	1327	**3470**
iphone 4s	3509	2412	**5921**
iphone 5	8627	8128	**16755**
iphone 5s	5779	4053	**9832**
kindle fire	1492	1311	**2803**
lenovo thinkpad	13998	9104	**23102**
mac mini	1350	1624	**2974**
macbook air	10291	6395	**16686**
macbook pro	20647	16049	**36696**
nexus 5	6054	5310	**11364**
nexus 7	2532	1629	**4161**
nexus 10	1962	1594	**3556**
nokia lumia 635	2042	2043	**4085**
samsumg galaxy tablet	488	569	**1057**
samsung galaxy note	788	1025	**1813**
samsung galaxy s4	6650	4806	**11456**
windows surface	1199	853	**2052**
Totals	**118727**	**90665**	**20939**

Language characteristics:

Location and device type do not explain the difference in messaging frequencies between control and treatment groups, but perhaps language does, especially if the test variation is available in one language or shows more fluency in one language, for instance. The following query should show this:

```
SELECT users.language,
   COUNT(*) AS count,
   experiments.experiment_group
 FROM tutorial.yammer_users users
 JOIN tutorial.yammer_experiments experiments
 ON users.user_id = experiments.user_id
 GROUP BY users.language, experiments.experiment_group
 ORDER BY users.language, experiments.experiment_group
```

By the pivot table below, it can again be seen that there is a reasonable split between control and test groups for this condition. While it is impossible to judge from this the linguistic quality of the test variation in all languages, it is at least the case that all languages seem represented in a fair enough capacity for this experiment.

	experiment_group		
language	control_group	test_group	**Totals**
arabic	62	31	**93**
chinese	55	33	**88**
english	907	439	**1346**
french	147	59	**206**
german	95	46	**141**
indian	56	22	**78**
italian	43	16	**59**
japanese	108	60	**168**
korean	29	15	**44**
portugese	41	23	**64**
russian	45	26	**71**
spanish	158	79	**237**
Totals	**1746**	**849**	**2595**

A glance at interaction types:

Next, we look at the frequencies of various interaction types between experimental groups and the software, using the following query:

```
SELECT events.event_name,
    COUNT(*) AS count,
    experiments.experiment_group
 FROM tutorial.yammer_events events
 JOIN tutorial.yammer_experiments experiments
 ON events.user_id = experiments.user_id
 GROUP BY events.event_name, experiments.experiment_group
 ORDER BY events.event_name, experiments.experiment_group
```

As with prior examinations, there is not much difference between experimental groups according to numbers of interactions each group collectively has experienced with the software, as shown by the following pivot table. However, for some reason there is a more striking imbalance between the two groups for the first few rows – the test group shows fewer than one-fifth as many users have entries for the "complete_signup", "create_user", "enter_email", and "enter_info" categories. Out of 1746 individuals in the control group, about 60% have these entries filled, while for the 849 users in the test group, only 24% do. It is not clear why this would have an impact, but it suggests we might look at more generic user-specific characteristics.

	experiment_group		
event_name	control_group	test_group	**Totals**
complete_signup	1055	203	**1258**
create_user	1055	203	**1258**
enter_email	1055	203	**1258**
enter_info	1055	203	**1258**
home_page	33553	26795	**60348**
like_message	20993	17097	**38090**
login	13970	9754	**23724**
search_autocomplete	6337	4338	**10675**
search_click_result_1	524	397	**921**
search_click_result_2	525	439	**964**
search_click_result_3	404	337	**741**
search_click_result_4	462	346	**808**
search_click_result_5	329	286	**615**

	experiment_group		
event_name	control_group	test_group	**Totals**
search_click_result_6	273	220	**493**
search_click_result_7	257	184	**441**
search_click_result_8	249	198	**447**
search_click_result_9	269	227	**496**
search_click_result_10	166	156	**322**
search_run	4666	3390	**8056**
send_message	11670	9748	**21418**
view_inbox	19860	15941	**35801**
Totals	**118727**	**90665**	**209392**

Account creation dates:

Among user characteristics from the users table, features that may relate to a user's history of interaction with the site include "created_at" date/time or "activated_at" date/time information. From the events table event types of "engagement" and "signup_flow" also exist and could be interesting, though it is not clear to what those terms refer or if they provide meaningful additional information beyond other characteristics to examine. Now, the following query will tell us about the history of creation of user accounts, allocated between control and test groups, in batches of time.

```
SELECT
  COUNT(CASE WHEN users.created_at BETWEEN '2013-01-01' AND '2013-04-01' THEN 1 ELSE NULL END) as
first_batch,
  COUNT(CASE WHEN users.created_at BETWEEN '2013-04-02' AND '2013-08-01' THEN 1 ELSE NULL END) as
second_batch,
  COUNT(CASE WHEN users.created_at BETWEEN '2013-08-02' AND '2013-12-31' THEN 1 ELSE NULL END) as
third_batch,
  COUNT(CASE WHEN users.created_at BETWEEN '2014-01-01' AND '2014-04-01' THEN 1 ELSE NULL END) as
fourth_batch,
  COUNT(CASE WHEN users.created_at BETWEEN '2014-04-02' AND '2014-07-01' THEN 1 ELSE NULL END) as
fifth_batch,
   experiments.experiment_group
 FROM tutorial.yammer_users users
 JOIN tutorial.yammer_experiments experiments
 ON users.user_id = experiments.user_id
 GROUP BY experiments.experiment_group ORDER BY experiments.experiment_group
```

Batches of time in the below table refer to these: first batch, between '2013-01-01' and '2013-04-01'; second batch, between '2013-04-02' and '2013-08-01'; third batch, between '2013-08-02' and '2013-12-31'; fourth batch, between '2014-01-01' and '2014-04-01'; and the fifth batch, between '2014-04-02' and '2014-07-01'. The results of this query into user account creation histories is interesting; the control and test groups are equally represented by users who have older accounts tracing back to early 2013. While both groups likely trace a growth in the overall numbers of users of the service over time, the control group shows an enormous number of users from the fifth batch of time, indicating a high representation of new user accounts filling well over half (65%) of the control group.

	control_group	test_group
first_batch	66	65
second_batch	95	108
third_batch	220	220
fourth_batch	224	163
fifth_batch	1134	288

A similar query for the “activated_at” entry for each batch in place of “created_at” produces identical results (not shown).

Conclusion:

The third hypothesis appears to be true; the allocation of users to control and test groups does not appear random, with a strong balance toward very new users within the control group. This very high prevalence of recently created accounts within the control group could explain the relatively lower frequency of messages sent by the group relative to the test group, as newer users may not be as familiar with or as dependent on the software for messaging purposes compared with users who have been around the software for a while.