**-----------SUMMARY ----------**

This exercise is part of the requirements of the [Springboard Data Science Career Track program](https://www.springboard.com/workshops/data-science-career-track/). It is one of the curated tutorials at [Mode Analytics](https://modeanalytics.com/) website. According to that website, their case study problems were drawn from suggestions and interviews of Analytics managers, and are meant to not only demonstrate critical analytical thinking ability using SQL syntax, but also allow one understand the meaning behind what is being measured.

I have taken this approach to show my understanding of SQL analytics. Although the requirement of the program is to complete at least one of the [cases](https://community.modeanalytics.com/sql/tutorial/sql-business-analytics-training/), i will try to do more as time permits. If you short of time, you can skip to the cases and/or problem, and my solution. By following by thought process, anyone should be able to replicate the same analysis. Thank you for reading this.

**-------CASE STUDY-----------**

**Investigating a Drop in User Engagement Problem**

Yammer’s Analysts are responsible for triaging product and business problems as they come up. In many cases, these problems surface through key metric dashboards that execs and managers check daily.

You show up to work Tuesday morning, September 2, 2014. The head of the Product team walks over to your desk and asks you what you think about the latest activity on the user engagement dashboards. You fire them up, and something immediately jumps out:

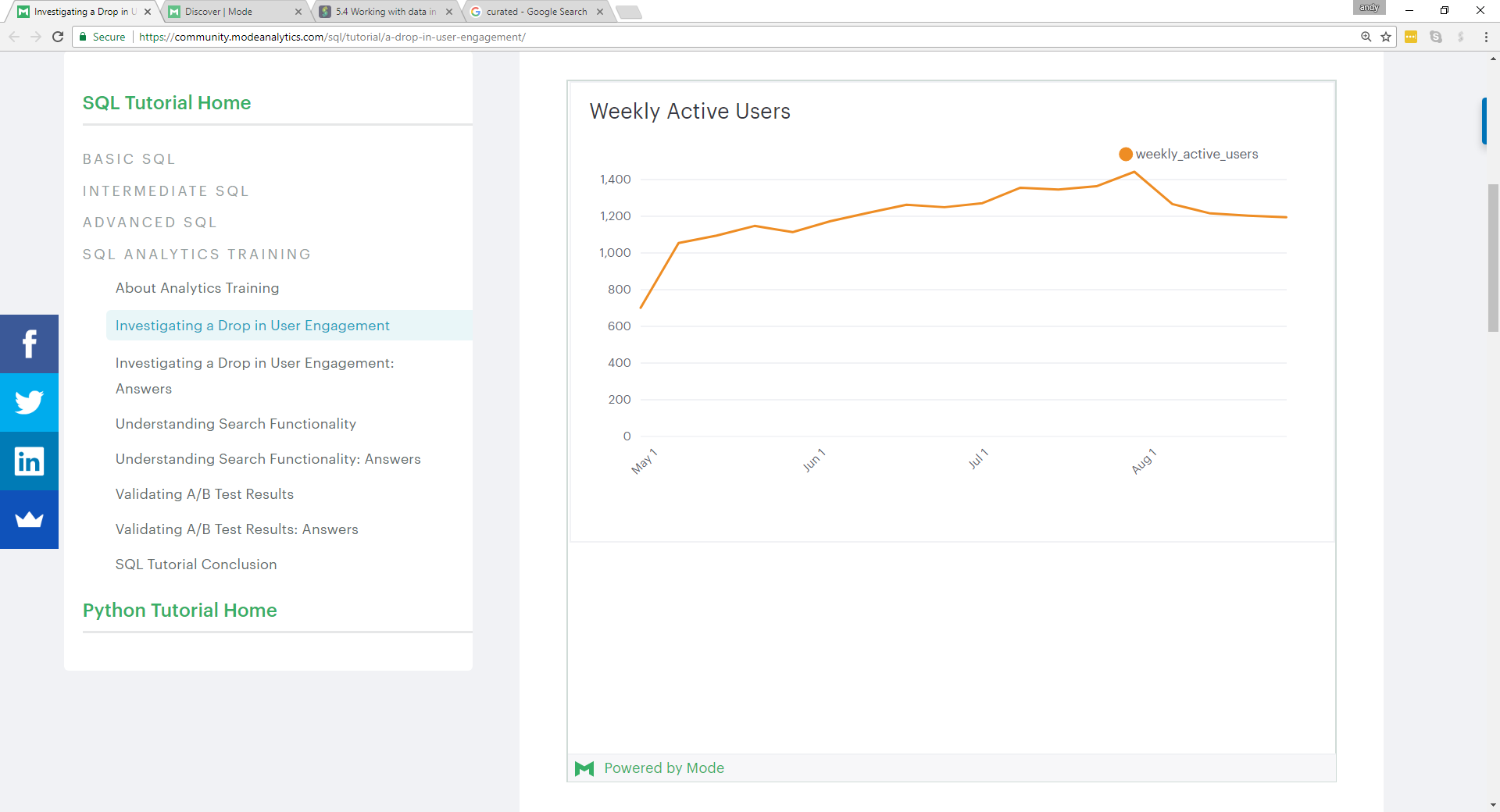


Figure 1: Chart adapted from <https://community.modeanalytics.com/sql/tutorial/a-drop-in-user-engagement/>

The above chart shows the number of engaged users each week. Yammer defines engagement as having made some type of server call by interacting with the product (shown in the data as events of type “engagement”). Any point in this chart can be interpreted as “the number of users who logged at least one engagement event during the week starting on that date.”

You are responsible for determining what caused the dip at the end of the chart shown above and, if appropriate, recommending solutions for the problem.

**Getting oriented**

Before you even touch the data, come up with a list of possible causes for the dip in retention shown in the chart above. Make a list and determine the order in which you will check them. Make sure to note how you will test each hypothesis. Think carefully about the criteria you use to order them and write down the criteria as well.

Also, make sure you understand what the above chart shows and does not show.

If you want to check your list of possible causes against ours, read the first part of the answer key.

**Digging in**

Once you have an ordered list of possible problems, it’s time to investigate.

For this problem, you will need to use four tables. The tables names and column definitions are listed below—click a table name to view information about that table. Note: this data is fake and was generated for the purpose of this case study. It is similar in structure to Yammer’s actual data, but for privacy and security reasons it is not real.

**Table 1: Users**

This table includes one row per user, with descriptive information about that user's account.

This table name in Mode is tutorial.yammer\_users

**user\_id:** A unique ID per user. Can be joined to user\_id in either of the other tables.

**created\_at:** The time the user was created (first signed up)

**state:** The state of the user (active or pending)

activated\_at: The time the user was activated, if they are active

company\_id: The ID of the user's company

language: The chosen language of the user

**Table 2: Events**

This table includes one row per event, where an event is an action that a user has taken on Yammer. These events include login events, messaging events, search events, events logged as user’s progress through a signup funnel, events around received emails.

This table name in Mode is tutorial.yammer\_events

user\_id: The ID of the user logging the event. Can be joined to user\\_id in either of the other tables.

occurred\_at: The time the event occurred.

event\_type: The general event type. There are two values in this dataset: "signup\_flow", which refers to anything occurring during the process of a user's authentication, and "engagement", which refers to general product usage after the user has signed up for the first time.

event\_name: The specific action the user took. Possible values include:

create\_user: User is added to Yammer's database during signup process

enter\_email: User begins the signup process by entering her email address

enter\_info: User enters her name and personal information during signup process

complete\_signup: User completes the entire signup/authentication process

home\_page: User loads the home page

like\_message: User likes another user's message

login: User logs into Yammer

search\_autocomplete: User selects a search result from the autocomplete list

search\_run: User runs a search query and is taken to the search results page

search\_click\_result\_X: User clicks search result X on the results page, where X is a number from 1 through 10.

send\_message: User posts a message

view\_inbox: User views messages in her inbox

location: The country from which the event was logged (collected through IP address).

device: The type of device used to log the event.

**Table 3: Email Events**

This table contains events specific to the sending of emails. It is similar in structure to the events table above.

This table name in Mode is tutorial.yammer\_emails

user\_id: The ID of the user to whom the event relates. Can be joined to user\_id in either of the other tables.

occurred\_at: The time the event occurred.

action: The name of the event that occurred. "sent\_weekly\_digest" means that the user was delivered a digest email showing relevant conversations from the previous day. "email\_open" means that the user opened the email. "email\_clickthrough" means that the user clicked a link in the email.

**Table 4: Rollup Periods**

The final table is a lookup table that is used to create rolling time periods. Though you could use the INTERVAL() function, creating rolling time periods is often easiest with a table like this. You won't necessarily need to use this table in queries that you write, but the column descriptions are provided here so that you can understand the query that creates the chart shown above.

This table name in Mode is benn.dimension\_rollup\_periods

period\_id: This identifies the type of rollup period. The above dashboard uses period 1007, which is rolling 7-day periods.

time\_id: This is the identifier for any given data point — it's what you would put on a chart axis. If time\_id is 2014-08-01, that means that is represents the rolling 7-day period leading up to 2014-08-01.

pst\_start: The start time of the period in PST. For 2014-08-01, you'll notice that this is 2014-07-25 — one week prior. Use this to join events to the table.

pst\_end: The start time of the period in PST. For 2014-08-01, the end time is 2014-08-01. You can see how this is used in conjunction with pst\_start to join events to this table in the query that produces the above chart.

utc\_start: The same as pst\_start, but in UTC time.

pst\_start: The same as pst\_end, but in UTC time.

**Making a recommendation**

Start to work your way through your list of hypotheses in order to determine the source of the drop in engagement. As you explore, make sure to save your work. It may be helpful to start with the code that produces the above query, which you can find by clicking the link in the footer of the chart and navigating to the “query” tab.

Answer the following questions:

Do the answers to any of your original hypotheses lead you to further questions?

If so, what are they and how will you test them?

If they are questions that you can’t answer using data alone, how would you go about answering them (hypothetically, assuming you actually worked at this company)?

What seems like the most likely cause of the engagement dip?

What, if anything, should the company do in response?

---------- MY THOUGHT PROCESS & CODES -----------

**Hypotheses**

What probably makes sense to me from the trend in that time series chart in Figure 1 are thus

1. **Seasonal hiring:** some companies hire lot of temporary workers during some seasons to keep up with peak times, and let the workers go out of season. That probably explains why the number of workers progressively increased and then dipped.
2. **Vacation:** This is reasonable because July and August are when most people take time off for vacations because the weather is warmer. So, users could somewhat reduce. However, it may not justify why user count remained steady after the dip because the numbers should be back up when people return from vacation.
3. **Outdoor activities:** summer is very warm depending on what part of the world you are at, and it’s usually time of the year you have outdoor activities. It probably may not make sense except if there are road warrior employees working remotely. In that case, warmer weather are very suitable for mobility. This takes into consideration that Yammer, the product in question, is an in-house social network for communicating with coworkers. However, work of remote employees depends on client engagements, and not time of the year. This is tricky, but just some thoughts that is a possibility
4. **Technical defect**: it is possible for parts of the product to have issues. If that is the case, then it is expected that the trend will go back up and levels out at the peak when the issue is reported and resolved. Question is, why did the trend level out after the deep? Did the company decide not to fix the issues?
5. **Usage dissatisfaction:** it could be that some users just stopped using it because they had issues with the product. Probably they gave it a try, and when it didn’t work the way they wanted, they just back out. That could explain the progressive increase in users and then the dip.

With these few hypotheses handy, we can then carry out some exploratory data analysis (EDA) to see if my thinking aligns, and check out the story the data tells. Asking questions like these before looking at the dataset at all makes analysis targeted.

Pulling up and join first three tables using the user\_id as the key with the code below:

*SELECT \**

*FROM tutorial.yammer\_users AS a, tutorial.yammer\_events AS b, tutorial.yammer\_emails AS c*

*WHERE a.user\_id = b.user\_id AND b.occurred\_at = c.occurred\_at*

*ORDER BY b.occurred\_at*

That gives us all columns required. I needed to order the occured\_at because the original chart was a time series from May to August. Out of the columns from that output, the only ones needed to meet the requirement for the case should be output from the code below:

*SELECT b.user\_id,*

*company\_id,*

*language,*

*b.occurred\_at,*

*event\_name,*

*location,*

*device,*

*action*

*FROM tutorial.yammer\_users AS a,*

*tutorial.yammer\_events AS b,*

*tutorial.yammer\_emails AS c*

*WHERE a.user\_id = b.user\_id*

*AND b.occurred\_at = c.occurred\_at*

*ORDER BY b.occurred\_at*

Now, we know there is an active user count dip on July 28, 2014 signified by the original plot of active users against the dates, but not why. Looking closely at the chart in Figure 1 and table from the SQL code above, we see that date and time are constant in this scenario across all variablesbecause we are looking at same , but we further investigate these reasons by drilling down into the user counts, which is a function of the user\_id. When i think users, look at the user\_id, i think what are the users doing, how are they doing it and where

Looking at the chart, we see that