



Calculating Churn Rate

Daniel Ade (3/19-5/19)

Table of Contents

1. Get familiar with Codeflix

- How many months has the company been operating? Which months do you have enough information to calculate a churn rate?
- What segments of users exist?

2. What is the overall churn trend since Codeflix started?

3. Compare the churn rates between user segments

1. Get Familiar with Codeflix

1.1 Get Familiar with Codeflix

- **How many months has the company been operating?**
 - Codeflix has been in operation for 4 months
12/01/16 through 03/31/17
- There is enough information to calculate churn rate for three months. This is determined from “2.” that shows subscription start at 12/1/16 until 03/30/17, where there are *no* cancellations in the first month of December.
- There are two segments of users that exist that we discover from the “1.” query:: 87 & 30.

Database Schema

subscriptions

id	subscription_start	subscription_end	segment
INTEGER	TEXT	TEXT	INTEGER

```
-- 1. Take a look at the first 100 rows of data in  
the --subscriptions table. How many different  
segments do --you see?
```

```
SELECT *  
FROM subscriptions  
LIMIT 100;
```

```
-- 2. Determine the range of months of data provided.  
-- Which months will you be able to calculate churn  
-- for?
```

```
SELECT MIN(subscription_start),  
MAX(subscription_start)  
FROM subscriptions;
```

1.2 Get Familiar with Codeflix (cont'd)

- It is important to create a months table with the three months we plan to calculate churn rate from in order to CROSS JOIN this with our subscriptions table. This will contribute to our analysis of determining in which months subscriptions were active and also canceled.
-

```
-- 4. Create a temporary table, cross_join, from
-- subscriptions and your months. Be sure to SELECT
-- every column.
```

```
desired_temp_table AS
(SELECT *
FROM subscriptions
CROSS JOIN months)
```

```
-- 3. You'll be calculating the churn rate for both
-- segments (87 and 30) over the first 3 months of
2017
-- (you can't calculate it for December, since there
-- are no subscription_end values yet). To get
started,
-- create a temporary table of months.
```

```
WITH months AS
(SELECT
    '2017-01-01' as first_day,
    '2017-01-31' as last_day
UNION
SELECT
    '2017-02-01' as first_day,
    '2017-02-28' as last_day
UNION
SELECT
    '2017-03-01' as first_day,
    '2017-03-31' as last_day
)
```

1.3 Get Familiar with Codeflix (cont'd)

- Creating the status table (see “5.”) will create a set of data that can be used to determine how many subscriptions were active at the start of the month. This will be our denominator in calculating churn rate.
- Calculating cancellations within each month is the next step to give us our numerator in aggregate calculations (see “6.”)

```
-- 6. Add an is_canceled_87 and an is_canceled_30
-- column to the status temporary table.
...
CASE
    WHEN segment = 30
    AND (subscription_start < first_day)
    AND (subscription_end BETWEEN first_day
    AND last_day) THEN 1
    ELSE 0
END as is_canceled_30,
CASE
    WHEN segment = 87
    AND (subscription_start < first_day)
    AND (subscription_end BETWEEN first_day
    AND last_day) THEN 1
    ELSE 0
END as is_canceled_87
FROM desired_temp_table)
```

```
-- 5. Create a temporary table, status, from the
-- cross_join table you created.

status AS
(SELECT id, first_day as 'month',
CASE
    WHEN segment = 87 AND (subscription_start
    < first_day)
    AND (subscription_end >= first_day
    OR subscription_end IS NULL
    ) THEN 1
    ELSE 0
END as is_active_87,
CASE
    WHEN segment = 30 AND (subscription_start
    < first_day)
    AND (subscription_end >= first_day
    OR subscription_end IS NULL
    ) THEN 1
    ELSE 0
END as is_active_30
FROM desired_temp_table)
```

2. What is the overall churn trend since Codeflix started?

2.1 What is the overall churn trend since Codeflix started?

- With our 7th objective as displayed, our goal is to calculate the sum of the active users of each particular segment and also the cancellations.
- We can use this data to calculate the churn rate over the three month period (see “8.”) [results in “Section 3”]
- To determine the *overall* churn rate, one would simply take the sum of cancel_87 & cancel_30 over the combined sum of both active segments (see “8.1”)

```
-- 8.1 Calculate the overall churn rate over the
three
-- month period.
```

```
SELECT month, 1.0 * (sum_canceled_87 +
    sum_canceled_30)/(sum_active_87 +
    sum_active_30)
    AS 'overall_churn'
FROM status_aggregate;
```

```
-- 7.Create a status_aggregate temporary table that
is
-- a SUM of the active and canceled subscriptions for
-- each segment, for each month.
```

```
status_aggregate AS
(SELECT month,
    SUM(is_active_87) as sum_active_87,
    SUM(is_active_30) as sum_active_30,
    SUM(is_canceled_87) as sum_canceled_87,
    SUM(is_canceled_30) AS sum_canceled_30
FROM status
GROUP BY month
)
```

```
-- 8. Calculate the churn rates for the two segments
-- over the three month period. Which segment has a
-- lower churn rate?
```

```
SELECT month,
    1.0 * sum_canceled_87/sum_active_87 AS
    '87_churn',
    1.0 * sum_canceled_30/sum_active_30 AS
    '30_churn'
FROM status_aggregate;
```


2.2 What is the overall churn trend since Codeflix started?

- From these results, we can see that overall churn rate increased each month. However, determining the churn rate between each segment will give us a better understanding of which group has better results (i.e. a lower churn rate).

Query Results	
month	overall_churn
2017-01-01	0.16140350877193
2017-02-01	0.188832487309645
2017-03-01	0.27164416203336

**3. Compare the churn rates
between user segments**

3.1 Compare the churn rates between user segments

- Based on the results of calculating the churn rate of segment 87 and 30, one can see that the churn rate for segment 30 is significantly lower for all three months.
- Based on these results, expanding segment 30 would be in the best interest of Codeflix.

Query Results		
month	87_churn	30_churn
2017-01-01	0.25089605734767	0.0756013745704467
2017-02-01	0.316916488222698	0.0733590733590734
2017-03-01	0.476894639556377	0.116991643454039

```
-- 8. Calculate the churn rates for the two segments
-- over the three month period. Which segment has a
-- lower churn rate?
```

```
SELECT month,
       1.0 * sum_canceled_87/sum_active_87 AS
       '87_churn',
       1.0 * sum_canceled_30/sum_active_30 AS
       '30_churn'
FROM status_aggregate;
```