



Codeflix User Churn

Analyze Data with SQL

Usage Insights Funnels

Sergio Garcia-Puga

18/01/2020

Table of Contents

1. Get familiar with the company
2. What is the overall churn trend since the company started?
3. Compare the churn rates between user segments.

1. Get familiar with the company

1. 1 Codeflix and the data

Codeflix, a streaming video startup, is interested in measuring their user churn rate. It's early on in the business and people are excited to know how the company is doing. The marketing department is particularly interested in how the churn compares between two segments of users.

They provide you with a dataset containing subscription data for users who were acquired through two distinct channels.

The dataset provided to you contains one SQL table, subscriptions. Within the table, there are 4 columns:

- id - the subscription id
- subscription_start - the start date of the subscription
- subscription_end - the end date of the subscription
- segment - this identifies which segment the subscription owner belongs to

1. 2 How many months has the company been operating?

We are checking the first subscription date and the last subscription date in order to see how long the company has been operating.

Which months do you have enough information to calculate a churn rate?

Codeflix requires a minimum subscription length of 31 days, so a user can never start and end their subscription in the same month. This means that we can only calculate the churn of January, February and March

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

MIN(subscription_start)	MAX(subscription_start)
2016-12-01	2017-03-30

1. 3 What segments of users exist?

The two segments are called 87 and 30. We will be comparing the churn rates of these user groups.

-

```
SELECT DISTINCT segment  
FROM subscriptions;
```

segment
87
30

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

2. 1 Churn rate

Our final goal is to compare the user churn rates for two segments of Codeflix users.

Churn rate is the percent of subscribers that have canceled within a certain period, usually a month. For a user base to grow, the churn rate must be less than the new subscriber rate for the same period.

To calculate the churn rate, we only will be considering users who are subscribed at the beginning of the month. The churn rate is the number of these users who cancel during the month divided by the total number:

$$\text{Churn rate} = \text{cancellations} / \text{active subscribers}$$

2.1 Calculating Churn in SQL

```
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  
) ,  
cross_join AS(  
    SELECT * FROM subscriptions  
    CROSS JOIN months  
) ,  
status AS(  
    SELECT id,  
    first_day AS month,  
    segment,  
    CASE WHEN  
        subscription_start < first_day
```

```
    AND (subscription_end > first_day or  
    subscription_end IS NULL) THEN 1  
        ELSE 0  
    END AS is_active,  
    CASE WHEN  
        (subscription_end BETWEEN first_day AND  
    last_day) THEN 1  
        ELSE 0  
    END AS is_canceled  
FROM cross_join),  
status_aggregate AS (  
    SELECT  
        month,  
        segment,  
        SUM(is_active) AS sum_active,  
        SUM(is_canceled) AS sum_canceled  
    FROM status  
    GROUP BY month  
)  
SELECT month,  
1.0 * sum_canceled / sum_active AS churn  
FROM status_aggregate;
```

2.2 Churn Trend

We see that the trend on the first 3 months is an increase in the churn, so looks like the company is getting an increased amount of cancellations over this period.

month	churn
2017-01-01	0.161687170474517
2017-02-01	0.189795918367347
2017-03-01	0.274258219727346

3. Compare the churn rates between user segments.

3.1 Calculate Churn per segment

```
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  
) ,  
cross_join AS(  
    SELECT * FROM subscriptions  
    CROSS JOIN months  
) ,  
status AS(  
    SELECT id,  
    first_day AS month,  
        segment,  
    CASE WHEN  
        subscription_start < first_day
```

```
    AND (subscription_end > first_day or  
        subscription_end IS NULL) THEN 1  
        ELSE 0  
        END AS is_active,  
    CASE WHEN  
        (subscription_end BETWEEN first_day AND  
        last_day) THEN 1  
        ELSE 0  
        END AS is_canceled  
    FROM cross_join),  
status_aggregate AS (  
    SELECT  
        month,  
        segment,  
        SUM(is_active) AS sum_active,  
        SUM(is_canceled) AS sum_canceled  
    FROM status  
    GROUP BY month, segment  
    ORDER BY segment  
)  
SELECT month, segment,  
    1.0 * sum_canceled / sum_active AS churn  
FROM status_aggregate;
```

2.2 Compare the churn rates between user segments.

After calculating the churn rates of both segments:

- We see that user churn for segment 87 is much higher than segment 30
- Both segments seem to have an increase in their user churn over the period, but it is especially alarming the increase in the user churn in segment 87 where almost 50% is leaving the company
- Although the churn trend is not good in segment 30, the percentage is much lower. On the last month only 11% unsubscribe from the services
- Additional data is needed in order to get a deeper analysis to understand the problem and decrease the user churn in future months

Query Results		
month	segment	churn
2017-01-01	30	0.0756013745704467
2017-02-01	30	0.0733590733590734
2017-03-01	30	0.11731843575419
2017-01-01	87	0.251798561151079
2017-02-01	87	0.32034632034632
2017-03-01	87	0.485875706214689