codecademy

Codeflix Subscription Churn

Learn SQL from Scratch
by Marshall Andrews

Table of Contents

1. Background on Codeflix subscriptions
2. Churn Rate by months
3. Churn Rates between segments
4. Conclusion

1. Background

Codeflix is a streaming video startup company. They began offering subscriptions through two distribution channels on December 01, 2016.

They have asked if their subscription data can be analyzed to determine their churn rate. Churn rate is the percent of subscribers that have canceled within a one month period. The churn rates for January/2017, February/2017, and March/2017 will be analyzed. Below is a sample of the subscription data that Codeflix has collected that will be used for analysis.

id	subscription_start	subscription_end	segment
1	2016-12-01	2017-02-01	87
2	2016-12-01	2017-01-24	87

2. Churn Rate by months

The churn rate is being calculated by using the number of users canceling during the month divided by the total number of users who are subscribed on the first day of each month.

month	churn_rate
2017-01-01	0.162
2017-02-01	0.19
2017-03-01	0.274

```
SQL CODE
-- Creates a temp file of 1st and last day of Jan, Feb, Mar 2017
-- called 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),
-- Creates a temp cross join file from subscriptions and months
cross_join AS
  (SELECT subscriptions.*, months.*
  FROM subscriptions
  CROSS JOIN months),
-- Creates a temp status file of active segment of 87 and 30
-- subscriptions
status AS
  (SELECT id, first_day as 'month',
   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),
-- Creates a temp status aggregate file
status_aggregate as
(SELECT month,
 sum(is_active) as 'active',
 sum(is_canceled) as 'canceled'
 FROM status
 GROUP BY month)
-- Creates a churn rate for each segment
SELECT month,
round(1.0 * canceled/active,3) AS 'churn_rate'
FROM status_aggregate;
```

3. Churn Rate by segment

Codeflix has two subscription distribution segments, 87 and 30. Each distribution segment’s churn rate is being calculated.

month	churn_rate _87	churn_rate _30	total_churn_rate
2017-01-01	0.252	0.076	0.162
2017-02-01	0.32	0.073	0.19
2017-03-01	0.486	0.117	0.274

```
-- SQL CODE
-- Creates a temp file of 1st and last day of Jan,Feb,Mar 2017 called 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),
-- Creates a temp cross join file from subscriptions and months
cross_join AS
  (SELECT subscriptions.*, months.*
  FROM subscriptions
  CROSS JOIN months),
-- Creates a temp status file of active segment of 87 and 30 subscriptions
status AS
  (SELECT id, first_day as 'month',
   CASE
   WHEN (subscription_start < first_day)
    AND (subscription_end > first_day
         OR subscription_end IS NULL)
    AND (segment = 87) THEN 1 ELSE 0
   END as is_active_87,
 CASE
   WHEN (subscription_start < first_day)
    AND (subscription_end > first_day
         OR subscription_end IS NULL)
    AND (segment = 30) THEN 1 ELSE 0
   END as is_active_30,
  CASE
  WHEN (subscription_end BETWEEN first_day AND last_day)
   AND (segment = 87) THEN 1 ELSE 0
   END as is_canceled_87,
  CASE
  WHEN (subscription_end BETWEEN first_day AND last_day)
   AND (segment = 30) THEN 1 ELSE 0
   END as is_canceled_30
   FROM cross_join),
-- Creates a temp status aggregate file
status_aggregate as
(SELECT month,
 sum(is_active_87) as 'active87',sum(is_active_30) as 'active30',
 sum(is_active_87) + sum(is_active_30) as 'total_active',
 sum(is_canceled_87) as 'canceled87', sum(is_canceled_30) as 'canceled30',
 sum(is_canceled_87) + sum(is_canceled_30) as 'total_canceled'
 FROM status
 GROUP BY month)
-- Creates a churn rate for each segment
SELECT month,
round(1.0 * canceled87/active87,3) AS 'churn_rate_87',
round(1.0 * canceled30/active30,3) AS 'churn_rate_30',
round(1.0 * total_canceled/total_active,3) AS 'total_churn_rate'
FROM status_aggregate;
```

4. Conclusion

Since Codeflix began offering subscription in December/2016, their churn rate has steadily increased.

For January/2017, their total churn was 16% and it has increased to 27 % by the end of March/2017, an 82% increase in churn. Within the distribution segments, segment 87’s churn increased from 25% to 48% for the same period, a 96% increase and segment 30’s churn increased from 8% to 12%, a 70% increase.

month	churn_rate_87	churn_rate_30	total_churn_rate
2017-01-01	0.252	0.076	0.162
2017-02-01	0.32	0.073	0.19
2017-03-01	0.486	0.117	0.274