codecademy

CODEFLIX

# USER CHURN RATE

Analyze Data with SQL

Bianca Niemann

02 September 2024

# Table of Contents

codecademy

# CODEFLIX

## 1a. Who is Codeflix?

Codeflix, a streaming video startup, and are interested in measuring their user churn rate.

They company started in Dec 2016 and they would like the churn rate for the first 3 months (Jan 2017 to Mar 2017), as cancellations are not possible within first 31 days of subscribing it would be pointless to analyze Dec 2016

## 1b. What is “Churn Rate”?

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.

$$\frac{\text{cancellations}}{\text{total subscribers}}$$

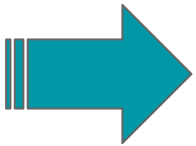
codecademy

CODEFLIX

# 2. Get familiar with the data

From “subscriptions” table, print first 20 rows

```
SELECT *
FROM subscriptions
LIMIT 20;
```



| id | subscription_start | subscription_end | segment |
|---|---|---|---|
| 1 | 2016-12-01 | 2017-02-01 | 87 |
| 2 | 2016-12-01 | 2017-01-24 | 87 |
| 3 | 2016-12-01 | 2017-03-07 | 87 |
| 4 | 2016-12-01 | 2017-02-12 | 87 |
| 5 | 2016-12-01 | 2017-03-09 | 87 |
| 6 | 2016-12-01 | 2017-01-19 | 87 |
| 7 | 2016-12-01 | 2017-02-03 | 87 |
| 8 | 2016-12-01 | 2017-03-02 | 87 |
| 9 | 2016-12-01 | 2017-02-17 | 87 |
| 10 | 2016-12-01 | 2017-01-01 | 87 |
| 11 | 2016-12-01 | 2017-01-17 | 87 |
| 12 | 2016-12-01 | 2017-02-07 | 87 |
| 13 | 2016-12-01 | | 30 |
| 14 | 2016-12-01 | 2017-03-07 | 30 |
| 15 | 2016-12-01 | 2017-02-22 | 30 |
| 16 | 2016-12-01 | | 30 |
| 17 | 2016-12-01 | | 30 |
| 18 | 2016-12-02 | 2017-01-29 | 87 |
| 19 | 2016-12-02 | 2017-01-13 | 87 |
| 20 | 2016-12-02 | 2017-01-15 | 87 |

## Determine the range of dates

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

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

## Determine the segments

```
SELECT DISTINCT(segment)
FROM subscriptions;
```

| segment |
|---|
| 87 |
| 30 |

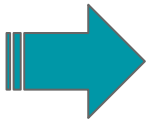
codecademy

CODEFLIX

# 3. Create Temp tables

# Create temp table called “months” that will have 2 columns “first_day” and “last_day”

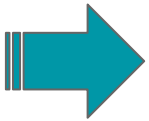
```
WITH months AS (
SELECT
        2017-01-01' AS 'first_day',
        2017-01-31' AS 'last_day'
UNION
SELECT
        2017-02-01' AS 'first_day',
        2017-02-28' AS 'last_day'
UNION
SELECT
        2017-03-01' AS 'first_day',
        2017-03-30' AS 'last_day'
),
```



| first_day | last_day |
|---|---|
| 2017-01-01 | 2017-01-31 |
| 2017-02-01 | 2017-02-28 |
| 2017-03-01 | 2017-03-30 |

## Create temp table “months” that will have 2 columns “first_day” and “last_day”

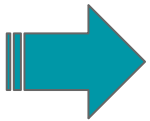
```
WITH months AS (
SELECT
        2017-01-01' AS 'first_day',
        2017-01-31' AS 'last_day'
UNION
SELECT
        2017-02-01' AS 'first_day',
        2017-02-28' AS 'last_day'
UNION
SELECT
        2017-03-01' AS 'first_day',
        2017-03-30' AS 'last_day'),
```



| first_day | last_day |
|---|---|
| 2017-01-01 | 2017-01-31 |
| 2017-02-01 | 2017-02-28 |
| 2017-03-01 | 2017-03-30 |

## Create temp table “cross_join” that will join “subscriptions” and “months” tables

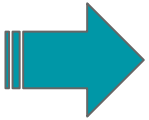
```
cross_join AS (
SELECT *
FROM subscriptions
CROSS JOIN months),
```



| id | subscription_start | subscription_end | segment | first_day | last_day |
|---|---|---|---|---|---|
| 1 | 2016-12-01 | 2017-02-01 | 87 | 2017-01-01 | 2017-01-31 |
| 1 | 2016-12-01 | 2017-02-01 | 87 | 2017-02-01 | 2017-02-28 |
| 1 | 2016-12-01 | 2017-02-01 | 87 | 2017-03-01 | 2017-03-30 |
| 2 | 2016-12-01 | 2017-01-24 | 87 | 2017-01-01 | 2017-01-31 |
| 2 | 2016-12-01 | 2017-01-24 | 87 | 2017-02-01 | 2017-02-28 |

```
status AS (
SELECT id,
   first_day AS 'month',
CASE
WHEN ( subscription_start < first_day)
AND (segment = 87)
AND (subscription_end > first_day
   OR subscription_end IS NULL)
THEN 1 ELSE 0
END AS is_active_87,
CASE
WHEN (subscription_start < first_day)
AND (segment = 30)
AND (subscription_end > first_day
   OR subscription_end IS NULL)
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_cancelled_87,
CASE
WHEN (subscription_end BETWEEN
first_day AND last_day)
AND (segment = 30)
THEN 1 ELSE 0
END AS is_cancelled_30
FROM cross_join),
```

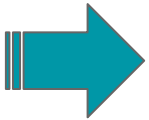
Create Temp table “status” showing “is_active” and “is_cancelled” columns for each segment



| id | month | is_active_87 | is_active_30 | is_cancelled_87 | is_cancelled_30 |
|---|---|---|---|---|---|
| 1 | 2017-01-01 | 1 | 0 | 0 | 0 |
| 1 | 2017-02-01 | 0 | 0 | 1 | 0 |
| 1 | 2017-03-01 | 0 | 0 | 0 | 0 |
| 2 | 2017-01-01 | 1 | 0 | 1 | 0 |
| 2 | 2017-02-01 | 0 | 0 | 0 | 0 |

## Create a “status_aggregate” temporary table that is a SUM of the active and canceled subscriptions for each segment, for each month

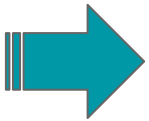
```
status_aggregate AS(
SELECT
month,
SUM(is_active_87) AS 'sum_active_87',
SUM(is_active_30) AS 'sum_active_30',
SUM(is_cancelled_87) AS
'sum_canceled_87',
SUM(is_cancelled_30) AS
'sum_canceled_30'
FROM status
GROUP BY month)
```



| month | sum_active_87 | sum_active_30 | sum_canceled_87 | sum_canceled_30 |
|---|---|---|---|---|
| 2017-01-01 | 278 | 291 | 70 | 22 |
| 2017-02-01 | 462 | 518 | 148 | 38 |
| 2017-03-01 | 531 | 716 | 247 | 81 |

## Calculate the churn rates for the two segments over the three month period.

```
SELECT
month,
ROUND(1.0 * sum_canceled_87 /
sum_active_87, 2) AS 'churn_rate_87',
ROUND(1.0 * sum_canceled_30 /
sum_active_30, 2) AS 'churn_rate_30'
FROM status_aggregate;
```

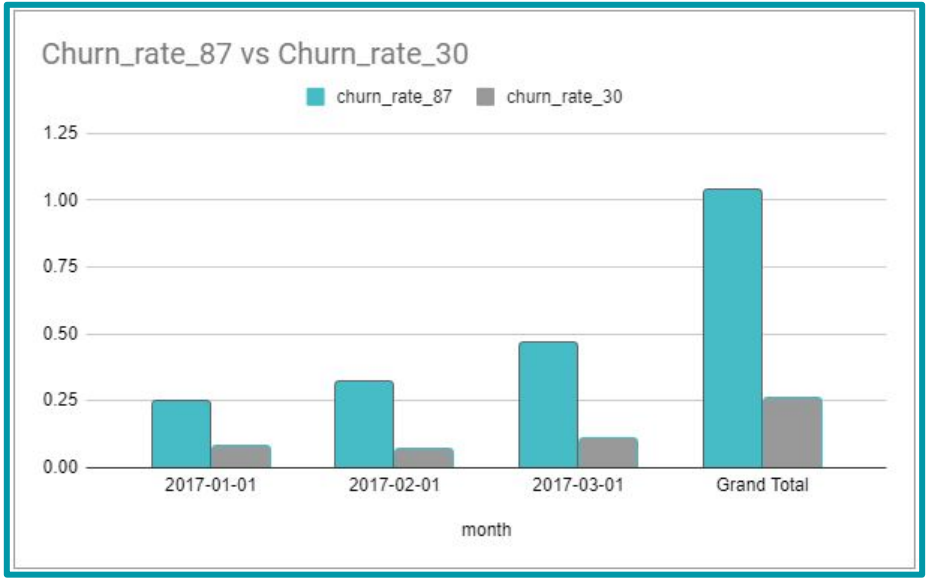


| month | churn_rate_87 | churn_rate_30 |
|---|---|---|
| 2017-01-01 | 0.25 | 0.08 |
| 2017-02-01 | 0.32 | 0.07 |
| 2017-03-01 | 0.47 | 0.11 |

codecademy

CODEFLIX

# 4. Conclusions




| month | churn_rate_87 | churn_rate_30 |
|---|---|---|
| 2017-01-01 | 0.25 | 0.08 |
| 2017-02-01 | 0.32 | 0.07 |
| 2017-03-01 | 0.47 | 0.11 |

- Based on our analysis, we can conclude that the churn rate increased over the months Jan 2017 to Mar 2017
- Customer segment 87 definitely has a much higher churn rate then customer segment 30
- Codeflix should look into why there is a general increase in Churn rates each month
- Codeflix should also see why 87 has a much higher rate and maybe make some changes to be more like 30 to decrease the Churn rate for that segment.