



# Churn Rate

Analyze Data with SQL

Sreeteja Pampati

5/2/2022

# Example Table of Contents

1. Introduction
2. Analysis
3. Appendix

# 1. Introduction

## 1.1 Codeflix

**Codeflix**, a streaming video startup, is interested in finding its churn rate between two segments of users. It requires a minimum subscription of a month. They are using an A/B test between two landing pages for the users.

Subscription Channel	Number of Users
87	1000
30	1000

month	first_day	last_day
01	2017-01-01	2017-01-31
02	2017-02-01	2017-02-28
03	2017-03-01	2017-03-31

```
SELECT DISTINCT segment, count(id)
FROM subscriptions
GROUP BY segment;
```

```
SELECT
strftime('%m', subscription_end) as 'month',
MIN(subscription_end) as first_day,
MAX(subscription_end) as last_day
FROM subscriptions
WHERE subscription_end IS NOT NULL
GROUP BY 1;
```

## 1.2 Churn Rate

**Churn rate** is a business metric used by SAS companies that measures the ratio of lost customers to active customers in a given period of time.

**It is useful** for analyzing how well a product or service is performing, and gives insight into why people are leaving.


$$\text{Churned (Leaving) Customers} \div \text{Total Customers} = \text{Churn Rate}$$



2 M



10 M



20%

## **2. Analysis**

## 2.1 Number of Users who Canceled

The number of users who left **Segment 87** was always greater than the number of users who left **Segment 30**. The rate of increase for users who left was also significantly larger for **Segment 87**.

Month	Left Segment 30	Left Segment 87
Jan	22	70
Feb	38	148
March	84	258

## 2.2 Churn Rate by Segment

**Churn Rate** for segment **87** increased from month to month. From January to February it increased by roughly 7%, and from February to March it increased by roughly 16.5%.

**Churn Rate** for segment **30** did not always increase from month to month. From January to February, it actually decreased by 0.22%, but then increased again by roughly 4.4%.

(Related Query is in the Appendix)

Month	Churn Rate Segment 30	Churn Rate Segment 87
Jan	7.56%	25.18%
Feb	7.34%	32.03%
March	11.73%	48.59%



## 2.3 Conclusions

Since the churn rate for Segment 87 is much larger than the churn rate for Segment 30, it is clear that Segment 30 is the higher performing segment. The company should focus on **expanding segment 30**.

What to do moving forward?

- Figure out why people are leaving for segment 87. Does it have to do with advertising - is the campaign targeting a relevant audience?
  - Company could talk to people in segment 87, identify key characteristics, and redevelop marketing campaigns for this audience.
- Analyze what is working so well with Segment 30, and implement these working strategies for the future
- Conduct follow up exercises to see if the trend continues past March. Perhaps segment 87 improves its churn rate as the year progresses.

## **3. Appendix**

# Churn Rate by Segment Code

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

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

SELECT
  month,
  1.0 *
  sum_canceled_30/sum_active_30 AS
  churn_rate_30,
  1.0 *
  sum_canceled_87/sum_active_87 AS
  churn_rate_87,
  sum_canceled_87,
  sum_canceled_30

FROM status_aggregate;
```