codecademy

Calculating Churn Rates

Analyze Data with SQL
Petru Apachitei
Date: 08-09-2020

Example Table of Contents

1. Introduction of the project
2. Churn rate
3. Compare churn rate between user segments and further insights

1. Introduction of the project

1.1 Introduction

What is the churn rate?
For SAS companies churn rate is a business metric for measuring how good your product is doing and refers to customers who have lost over a period of time.

How we calculate churn rate?
For this type of business the churn rate is calculated by the following mathematical ratio between number of users who canceled the subscription in a given time period divided by the total active users at the beginning of that period

What is helpful churn rate?
- See how your product is performing
- See the health of your product, why people are leaving, first step to diagnose why a product is doing well or not.

1.2 Company and dataset introduction

Company
- Codeflix, a streaming video startup, is interested in measuring the churn rate between two segments of users. **Codeflix requires a minimum of subscriptions length of 31 days.**

Dataset
- Dataset provided contain four columns, due to the minimum subscription length of 31 days, **there are 3 months for churn calculation**
- Codeflix running ad campaigns and create A/B test, **segmenting users by two landing page.**

Subscribtion channel	Number of users
30	1000
87	1000

Dataset time period
2017-01-01 2017-01-31
2017-02-01 2017-02-28
2017-03-01 2017-03-31

```sql
-- get familiar with the dataset.

SELECT *
 FROM subscriptions
 LIMIT 100;

-- extract from dataset the range of months
data provided

 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;

-- create a query to display  users segments

SELECT
  segment,
  COUNT(*)
FROM subscriptions
GROUP BY 1;
```
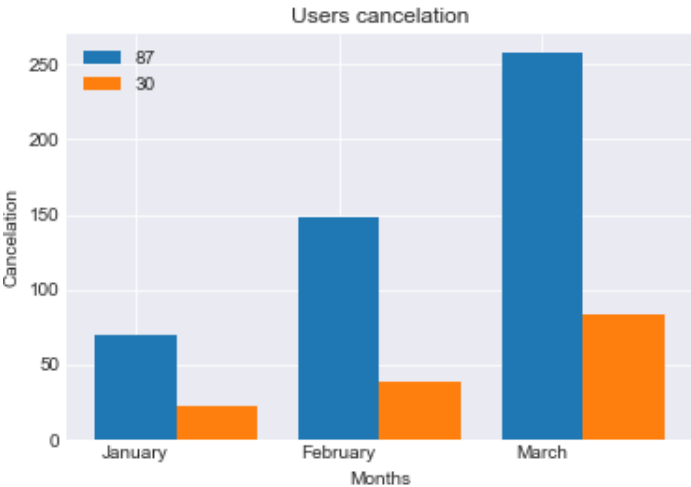
2. Churn rate

2. 1 Calculate the users who cancel the subscription

Calculate the users who left Codeflix for the time period provided by the dataset.
The related query is attached under Appendix 1.

Month	Left_segment_87	Left_segment_30
January	70	22
February	148	38
March	258	84

Users cancelation

# 3. Compare churn rate between user segments and further insights

3. 1 Calculate customer churn

Calculate churn rate for segment 87 and 30 for the three months period.
- January segment 87: 25% & segment 30 : 8%
- January segment 87: 32% & segment 30 : 7%
- January segment 87: 49% & segment 30 : 12%

The related query is attached under Appendix 1.

Month	Churn_rate_segment_87(%)	Churn_rate_segment_30(%)
January	25.2	7.6
February	32.0	7.3
March	48.6	11.7
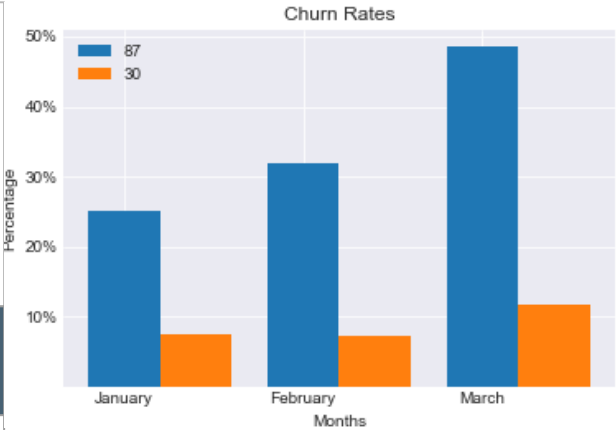
3.2 Conclusion

From the calculation results user from segment 87 has higher churn rate than segment 30, means that segment 30 has better performance.

What can we do for segment 87?

Why are they leaving?

- Go out and enggage with customers, talk to them, look at their proffile, identify characteristics and analyze past interactions with the product.

What action can we derive from the results?

- Call customers who left and ask them for feedback, communicate to them the latest developments that might be in their interest. This approach will create a felling of customers needs and perspective of your product. Preferable is calling and talk to them rather than send him to fill in a survey.

Which features have the most impact on a customer leaving?

This question should be addressed to customer support team.

Appendix 1

```
- 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
  FROM subscriptions),

-- Create a temporary table, cross_join, from
subscriptions and your months
cross_join as
  (SELECT *
   FROM subscriptions
   CROSS JOIN months
   ),
-- Create a temporary table, status, from the
cross_join table
status as
  (SELECT
    segment,
    id,
    first_day as month,
```

```
CASE
    WHEN (subscription_start < first_day AND
      ((subscription_end IS NULL) OR
(subscription_end > first_day))
      )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),
--SELECT * FROM status LIMIT 100;
-- 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,
    segment,
    SUM(is_active) as sum_active,
    SUM(is_canceled) as sum_canceled
  FROM status
  GROUP BY 1, 2)
--SELECT * FROM status_aggregate;
-- Calculate the churn rates for the two segments
over the three month period.
SELECT month,
  segment,
  ROUND (100.0 * (sum_canceled) / (sum_active), 1)
as churn_rate
FROM status_aggregate;
```