codecademy

# Capstone: Churn Rates

By Yoav Schatzberg

# Example Table of Contents

1. Get familiar with Codeflix
2. What is the overall churn rate?
3. Compare the churn rates between segments and months

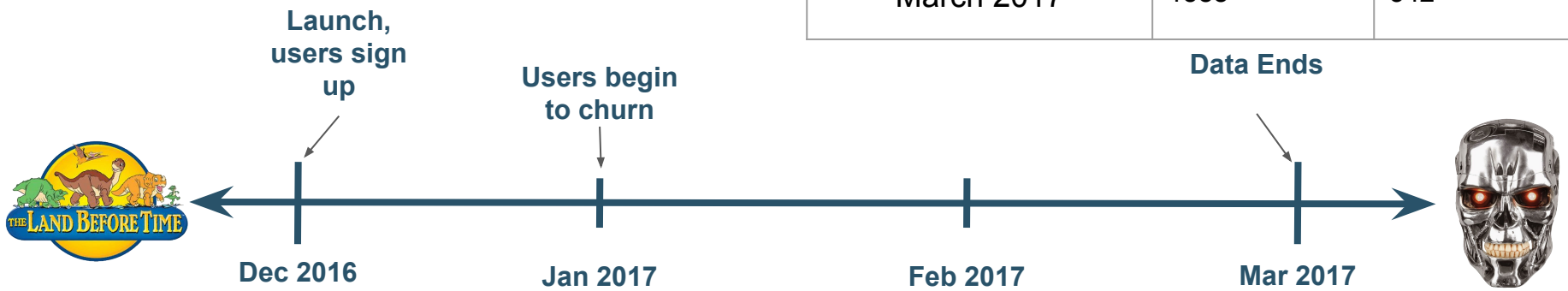
# Getting Familiar With Codeflix

# Codeflix, a Brief History

Data started being sent for Codeflix in December of 2016. However since there is a 1 month commitment, no churn occurred in December. From January to March 2017, we have Churn data. There is no data after March 2017

```
select month, SUM(is_active), SUM(is_canceled) from
status group by month order by month;
```

| Month | Active | Canceled |
|---|---|---|
| December 2016 | 17 | 0 |
| January 2017 | 582 | 92 |
| February 2017 | 1091 | 186 |
| March 2017 | 1588 | 342 |



# Segmenting the Code(flix)

There is an even number of users from each segment.

| User Segment | # of users |
|---|---|
| 30 | 1000 |
| 87 | 1000 |

```
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
),
...
status_aggregate as (
select
  segment,
  SUM(is_active) as sum_active,
  SUM(is_canceled) as sum_canceled
from
  status
)

select segment, count(*) from subscriptions group by
segment order by segment;
```

# Lets Get Churning

# Churn Rate Overall

**The overall Churn rate over time is**

**22%**

```
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
),
...
status_aggregate as (
select
  segment,
  SUM(is_active) as sum_active,
  SUM(is_canceled) as sum_canceled
from
  status
)

select Round(((sum_canceled*1.0)/sum_active),2) as
churn from status_aggregate group by segment order by
segment;
```

# Churn Rate by Segment

- Looking at the churn rate by segment, it's clear that there's much higher retention in Segment 30. Segment 30 has a 9% churn rate, whereas segment 87 has a 30% churn rate (see what I did there?)
- As a company Codeflix will generate longer lasting customers, and ultimately more, by **focusing on segment 30 for expansion**.
- Someone should also look at **why the churn in segment 87 is so high**

| Segment | Churn Rate |
|---|---|
| 30 | 9% |
| 87 | 30% |

```
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
),
...
status_aggregate as (
select
  segment,
  SUM(is_active) as sum_active,
  SUM(is_canceled) as sum_canceled
from
  status group by segment order by segment
)

select segment,
Round(((sum_canceled*1.0)/sum_active),2) as churn
from status_aggregate group by segment order by
segment;
```

# Bonus (Appendix)

# Churn Rate by Month

There is an increasing trend in churn. Something to be wary about going forward.

| Month | Churn Rate |
|---|---|
| December 2016 | 0% |
| January 2017 | 16% |
| February 2017 | 17% |
| March 2017 | 22% |

```
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
),
...
status_aggregate as (
select
  month,
  SUM(is_active) as sum_active,
  SUM(is_canceled) as sum_canceled
from
  status group by month order by month
)

select month,
Round(((sum_canceled*1.0)/sum_active),2) as churn
from status_aggregate group by month order by month;
```

# Churn Rate by Month by Segment!

| Month/Segment | 30 | 87 |
|---|---|---|
| | | |
| January 2017 | 7% | 25% |
| February 2017 | 7% | 27% |
| March 2017 | 11% | 33% |

```
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
...
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
month,segment
)
select month, segment,
Round(((sum_canceled*1.0)/sum_active),2) as churn
from status_aggregate group by month,segment order by
month,segment;
```