# Project3
# Lifetime Value for drivers

**Author:** Jenny Huang
**Date:** 2023-07-02

AI LAUNCH LAB

# Contents

# 01 What is Lifetime Value?

# Amount of Revenue Generated by a driver

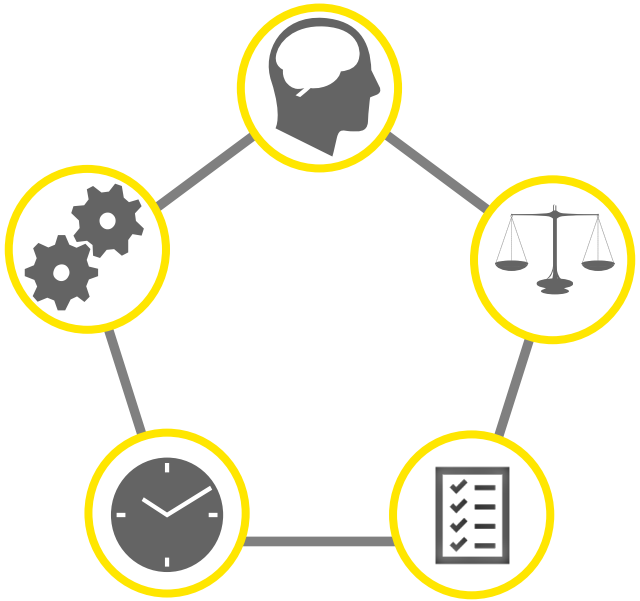
**Function**

You may think of this as primarily a function of operations, but those in recruiting can have a significant effect. Put simply, the more efficiently companies bring in quality drivers, the more money the companies make.

With Lifetime Value in mind, companies could change their focus to value and build their recruiting around producing revenue rather than cutting costs.

**Why should the company care?**

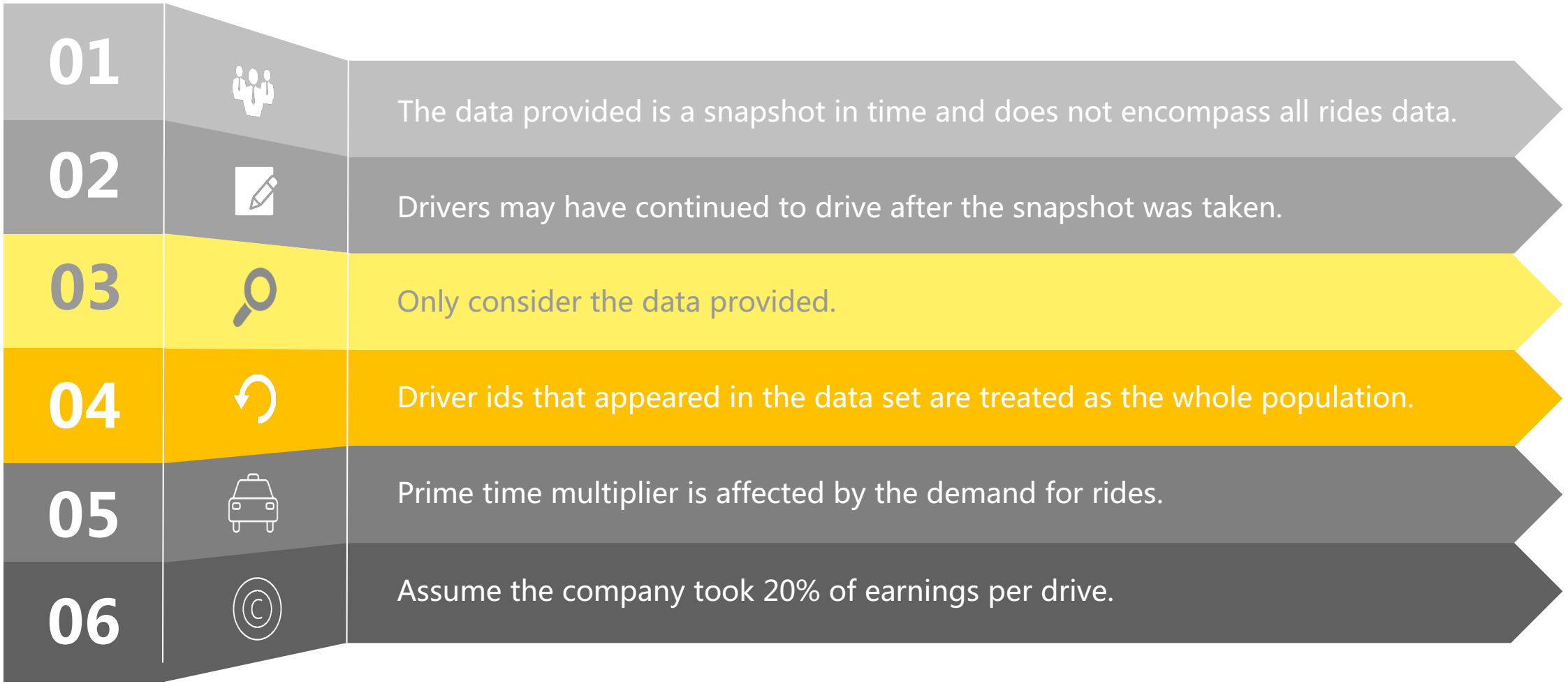
- Average lifetime value should be relavant to the company no matter market conditions, number of empty trucks, or current turnover rate.
- Average lifetime value changes the way the company look at metrics such as cost-per-hire and cost-per-approved application.
- A cost focus will force the company to monitor metrics that aren't predictive of revenue.



- With LTV in mind will help the company focus to build their recruiting around producing revenue rather than cutting costs.
- The company's focus should be to increase the amount of value they gets from the drivers they hire.
- This is the most important metric for recruiting that the company can track.

# 02 Assumptions

# Assumptions



# 03 Data set

# Overview of the data set

Firstly, **driver_ids.csv** contains fields **driver_id**, **driver_onboard_date**; Secondly, **ride_ids.csv** contains fields **driver_id**, **ride_id**, **ride_distance**, **ride_duration**, **ride_prime_time**; Thirdly, **ride_timestamps.csv** contains fields **ride_id**, **event**, **timestamp**. All of the above fields will be explained later.

| | driver_id | driver_onboard_date |
|---|---|---|
| 0 | 002be0ffdc997bd5c50703158b7c2491 | 2016-03-29 |
| 1 | 007f0389f9c7b03ef97098422f902e62 | 2016-03-29 |
| 2 | 011e5c5dfc5c2c92501b8b24d47509bc | 2016-04-05 |
| 3 | 0152a2f305e71d26cc964f8d4411add9 | 2016-04-23 |
| 4 | 01674381af7edd264113d4e6ed55ecda | 2016-04-29 |
| ... | ... | ... |
| 932 | ff419a3476e21e269e340b5f1f05414e | 2016-04-26 |
| 933 | ff714a67ba8c6a108261cd81e3b77f3a | 2016-03-28 |
| 934 | fff482c704d36a1afe8b8978d5486283 | 2016-04-08 |
| 935 | fffecccc49436c5389075b13209f0dfa | 2016-05-06 |
| 936 | ffff51a71f2f185ec5e97d59dbcd7a78 | 2016-05-04 |

# Overview of the data set

Firstly, **driver_ids.csv** contains fields **driver_id**, **driver_onboard_date**; Secondly, **ride_ids.csv** contains fields **driver_id**, **ride_id**, **ride_distance**, **ride_duration**, **ride_prime_time**; Thirdly, **ride_timestamps.csv** contains fields **ride_id**, **event**, **timestamp**. All of the above fields will be explained later.

| | driver_id | ride_id | ride_distance | ride_duration | ride_prime_time |
|---|---|---|---|---|---|
| 0 | 002be0ffdc997bd5c50703158b7c2491 | 006d61cf7446e682f7bc50b0f8a5bea5 | 1811 | 327 | 50 |
| 1 | 002be0ffdc997bd5c50703158b7c2491 | 01b522c5c3a756fbdb12e95e87507eda | 3362 | 809 | 0 |
| 2 | 002be0ffdc997bd5c50703158b7c2491 | 029227c4c2971ce69ff2274dc798ef43 | 3282 | 572 | 0 |
| 3 | 002be0ffdc997bd5c50703158b7c2491 | 034e861343a63ac3c18a9ceb1ce0ac69 | 65283 | 3338 | 25 |
| 4 | 002be0ffdc997bd5c50703158b7c2491 | 034f2e614a2f9fc7f1c2f77647d1b981 | 4115 | 823 | 100 |
| ... | ... | ... | ... | ... | ... |
| 193497 | ffff51a71f2f185ec5e97d59dbcd7a78 | fc717192b3512767269ff5a54b97af05 | 10127 | 1336 | 0 |
| 193498 | ffff51a71f2f185ec5e97d59dbcd7a78 | fd6fa5f9265d2cf83936ead663f9e0e7 | 1908 | 445 | 0 |
| 193499 | ffff51a71f2f185ec5e97d59dbcd7a78 | fe0857c43025264d337dfe1d8463e503 | 4039 | 875 | 0 |
| 193500 | ffff51a71f2f185ec5e97d59dbcd7a78 | ff0db0ca4557bf5b05b4da6f660a1ac1 | 4760 | 777 | 0 |
| 193501 | ffff51a71f2f185ec5e97d59dbcd7a78 | ff7dc29693f8c79ff103d350a7b6c157 | 3751 | 889 | 100 |

# Overview of the data set

Firstly, **driver_ids.csv** contains fields **driver_id**, **driver_onboard_date**; Secondly, **ride_ids.csv** contains fields **driver_id**, **ride_id**, **ride_distance**, **ride_duration**, **ride_prime_time**; Thirdly, **ride_timestamps.csv** contains fields **ride_id**, **event**, **timestamp**. All of the above fields will be explained later.

| | ride_id | event | timestamp |
|---|---|---|---|
| 0 | 00003037a262d9ee40e61b5c0718f7f0 | requested_at | 2016-06-13 09:39:19 |
| 1 | 00003037a262d9ee40e61b5c0718f7f0 | accepted_at | 2016-06-13 09:39:51 |
| 2 | 00003037a262d9ee40e61b5c0718f7f0 | arrived_at | 2016-06-13 09:44:31 |
| 3 | 00003037a262d9ee40e61b5c0718f7f0 | picked_up_at | 2016-06-13 09:44:33 |
| 4 | 00003037a262d9ee40e61b5c0718f7f0 | dropped_off_at | 2016-06-13 10:03:05 |
| ... | ... | ... | ... |
| 970400 | ffffccd77f47a3de26dfed9a851464b4 | requested_at | 2016-05-18 08:44:13 |
| 970401 | ffffccd77f47a3de26dfed9a851464b4 | accepted_at | 2016-05-18 08:44:21 |
| 970402 | ffffccd77f47a3de26dfed9a851464b4 | arrived_at | 2016-05-18 08:44:36 |
| 970403 | ffffccd77f47a3de26dfed9a851464b4 | picked_up_at | 2016-05-18 08:44:42 |
| 970404 | ffffccd77f47a3de26dfed9a851464b4 | dropped_off_at | 2016-05-18 09:11:37 |

# Three CSV files (Raw data set)

**Data Assumptions:**

- All rides in the data set occurred in San Francisco;
- All timestamps in the data set are in UTC;
- This is a snapshot of onboarding and ride history data for a 3 month period;
- The data is complete for these drivers during the given time period;
- However, additional rides may have occurred before and after the time period included in the data;

## Driver_ids.csv

- Driver_id: Unique identifier for a driver;
- Driver_onboard_date: Date the driver was approved to drive with the company;
- **Value number: 937**

## Ride_timestamps.csv

- Ride_id: Unique identifier for a ride;
- Event: Event describes the type of event;
- Timestamp: Time of event;
- **Value number: 970405**

## Ride_ids.csv

- Driver_id: Unique identifier for a driver;
- Ride_id: Unique identifier for a ride that was completed by the driver;
- Ride_distance: Ride distance in meters;
- Ride_duration: Ride durations in seconds;
- Ride_prime_time: PrimeTime multiplier (%) applied on the ride;
- **Value number: 193502**

# Data info & Preparation

## Overview of the event types

- Requested_at: passenger requested a ride;
- Accepted_at: driver accepted a passenger request;
- Arrived_at: driver picked up the passenger;
- Pickup_at: driver picked up the passenger;
- Dropped_off_at: driver dropped off a passenger at destination;

## Overview of the rate card

| | | |
|---|---|---|
| **Base Fare:** $2.00 | **Cost per minute:** $0.22 | **Minimum fare:** $5.00 |
| **Cost per mile:** $1.15 | **Service fee:** $1.75 | **Maximum fare:** $400.00 |

**Way to calculate drivers' churn rate:**

I. Take a look at the whole data set and found 4 months in total which are 2016/3; 2016/4; 2016/5; 2016/6.
II. Use *driver_id* as key variable to compare with the appearance last month and get the churn rate for the neighbor month which will get 3 results.
III. Calculate the average of 3 results from above as final average churn rate.

**Way to calculate average projected lifetime of a driver:**

I. There are 3 months that contain drivers' onboard date and 4 months contain drivers' ride info.
II. Use 2016/3 as the standard point to compare the *driver_ids* in the following months.
III. For those *driver_ids* that appear in March but not in any of the following months, treat this driver quit in this month.

# 04 Data cleaning

# Data cleaning

## Check null values

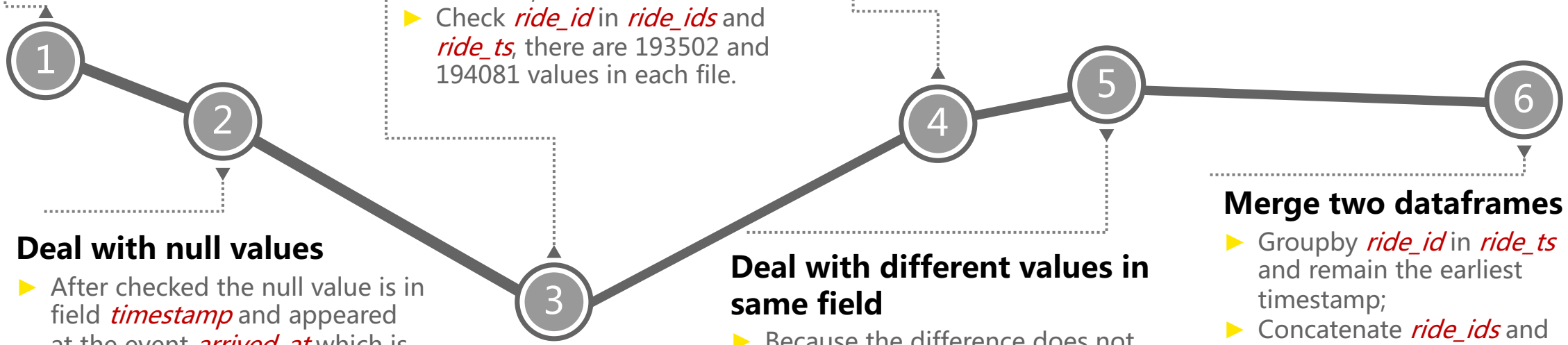
- As we can conclude from the info section of dataframe, *driver_ids* and *ride_ids* don't have any null values, but *ride_ts* has one.

## Unique values

- Check *driver_id* field's unique value in *driver_ids* and *ride_ids*, there are 937 *driver_id* values in each file;
- Check *ride_id* in *ride_ids* and *ride_ts*, there are 193502 and 194081 values in each file.

## Check the same fields for different dataset

- The intersection of *driver_id* in *driver_ids* and *ride_ids* is 854 values, so there are 64 values are different;
- The intersection of *ride_id* in *ride_ids* and *ride_ts* is 184819 values, so there are 8683 values in *ride_ids* are not in *ride_ts* and 9262 values in *ride_ts* are not in *ride_ids*.




## Deal with null values

- After checked the null value is in field *timestamp* and appeared at the event *arrived_at* which is basically the same, not technically, as the event *picked_up_at*, so we replace the timestamp at pick up time.

## Deal with different values in same field

- Because the difference does not matter when calculating churn rate and lifetime value, just ignore the difference and will take the intersection of *driver_id* for churn rate.

## Merge two dataframes

- Groupby *ride_id* in *ride_ts* and remain the earliest timestamp;
- Concatenate *ride_ids* and *ride_ts* on *ride_id* as a new dataframe called *ride_ids_v2*.

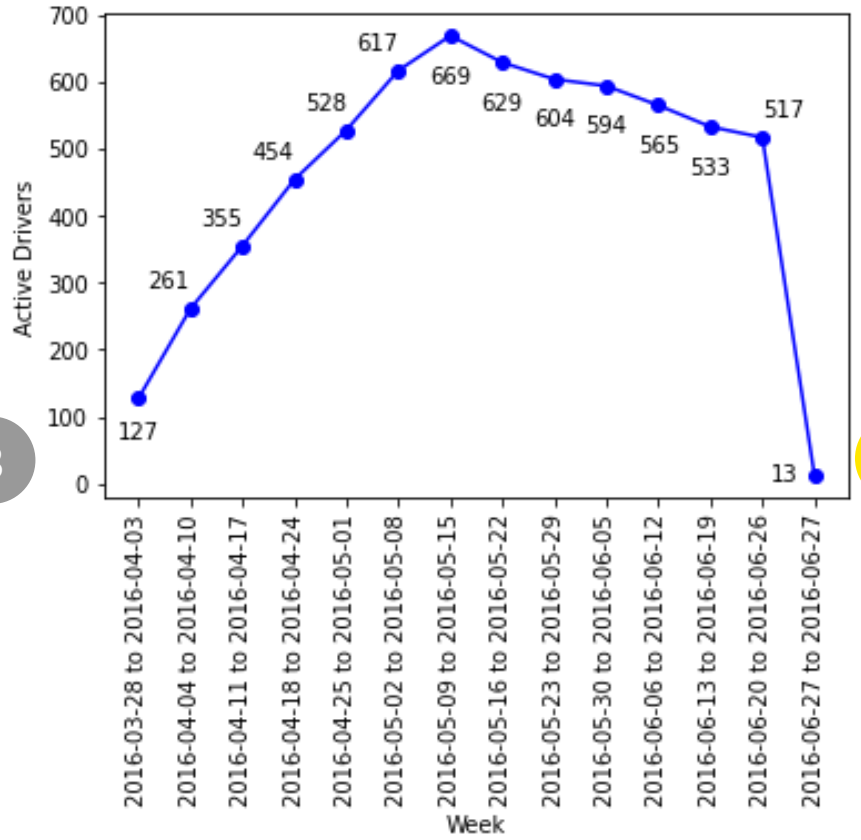
# 05 Exploratory data analysis (EDA)

# Active drivers over time (weekly basis)

1

## Method

Since we've combined the timestamp field into ***ride_ids***, we group by the ***timestamp*** of ***ride_ids_v2*** on a weekly basis and it turns out has 14 weeks in total, so we created 14 data frames to store the data separately and found the unique drivers in each data frame which were considered as active drivers on weekly basis.




2

## Numerical results:

- Week1: 127
- Week2: 261
- Week3: 355
- Week4: 454
- Week5: 528
- Week6: 617
- Week7: 669
- Week8: 629
- Week9: 604
- Week10: 594
- Week11: 565
- Week12: 533
- Week13: 517
- Week14: 13

3

## Date distribution:

- Week1: 2016-03-28 to 2016-04-03
- Week2: 2016-04-04 to 2016-04-10
- Week3: 2016-04-11 to 2016-04-17
- Week4: 2016-04-18 to 2016-04-24
- Week5: 2016-04-25 to 2016-05-01
- Week6: 2016-05-02 to 2016-05-08
- Week7: 2016-05-09 to 2016-05-15
- ……

4

## Date distribution (continue):

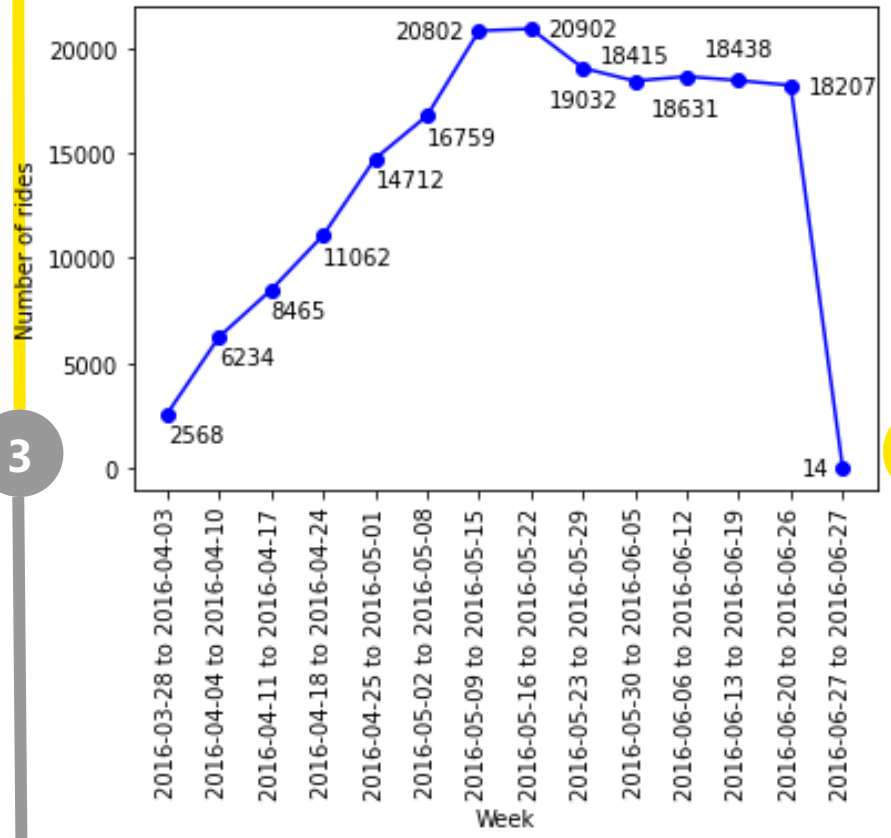
- Week8: 2016-05-16 to 2016-05-22
- Week9: 2016-05-23 to 2016-05-29
- Week10: 2016-05-30 to 2016-06-05
- Week11: 2016-06-06 to 2016-06-12
- Week12: 2016-06-13 to 2016-06-19
- Week13: 2016-06-20 to 2016-06-26
- Week14: 2016-06-27 to 2016-06-27

# Number of rides over time (weekly basis)

1

## Method

Same as before, group by the timestamp in ***ride_ids_v2*** on weekly basis and find unique values of ***ride_id*** in each week as the number of rides on a weekly basis.




2

## Numerical results:

- Week1: 2568
- Week2: 6234
- Week3: 8465
- Week4: 11062
- Week5: 14712
- Week6: 16759
- Week7: 20802
- Week8: 20902
- Week9: 19032
- Week10: 18415
- Week11: 18631
- Week12: 18438
- Week13: 18207
- Week14: 14

3

## Date distribution:

- Week1: 2016-03-28 to 2016-04-03
- Week2: 2016-04-04 to 2016-04-10
- Week3: 2016-04-11 to 2016-04-17
- Week4: 2016-04-18 to 2016-04-24
- Week5: 2016-04-25 to 2016-05-01
- Week6: 2016-05-02 to 2016-05-08
- Week7: 2016-05-09 to 2016-05-15
- .....

4

## Date distribution (continue):

- Week8: 2016-05-16 to 2016-05-22
- Week9: 2016-05-23 to 2016-05-29
- Week10: 2016-05-30 to 2016-06-05
- Week11: 2016-06-06 to 2016-06-12
- Week12: 2016-06-13 to 2016-06-19
- Week13: 2016-06-20 to 2016-06-26
- Week14: 2016-06-27 to 2016-06-27

# The correlation between drivers and rides number

## Method

**We used the same weekly drivers' number and rides' number that we got previously. If we plot the raw data, we cannot see any obvious trends from the figure, so we tried to plot the log scaled number values and it turns out they have a very close relationship to each other. We also calculate their correlation to prove the close relation.**



| | drivers | rides |
|---|---|---|
| drivers | 1.00 | 0.97 |
| rides | 0.97 | 1.00 |

## Result

**The correlation between drivers and rides is 0.97 which is highly positively related with each other. Each driver can only take rides at a certain level, so the number each drivers take should be similar.**

# Relationship between drivers/rides and prime time

## Introduction

**Previously, we have calculated the correlation between drivers and rides which is 0.97. Here, we will try to find out how the demand for rides impact the number of drivers base on the data we have. We already assume that the prime-time multiplier is affected by the demand for rides. In economics, it's called 'excess demand' which means the quantity demanded is greater than the quantity supplied at the given price so that the company will implement a prime-time charge to increase the price and lower the demand.**

| Data | Step I | Step II | Step III | Result analysis |
|---|---|---|---|---|
| We use *ride_ids* data frame to calculate the correlation between the PrimeTime multiplier and the drivers/rides | Check the unique value that appeared in the *ride_prime_time* field which are: 0, 25, 50, 75, 100, 150, 200, 250, 300, 350, 400, and 500 | Calculate the correlation between *ride_prime_time* and *ride_id* which is -0.56. Also, we have testified that their relationship fits exponential distribution (p-value=0.0). | Calculate the correlation between *ride_prime_time* and *driver_id* which is -0.94. Also, we have testified that their relationship fits normal distribution (p-value=1.24e-113). | ➢ From **Step II**, we could tell that the less the rides occur, the more the prime-time multiplier will be which means people will pay more for their rides when the demand is exceed the supply.<br>➢ From **Step III**, we could tell that the more the drivers, the less the prime-time multiplier will be. When the drivers' number is low, the prime-time price will get more expensive and vice versa.<br>➢ **Conclusion**: When the price gets higher, it will attract more drivers to provide more rides and after the supply-demand ratio getting bigger, the price will be pushed lower again. |

# Relationship between drivers/rides and prime time

## Introduction

**Previously, we have calculated the correlation between drivers and rides which is 0.97. Here, we will try to find out how the demand for rides impact the number of drivers base on the data we have. We already assume that the prime-time multiplier is affected by the demand for rides. In economics, it's called 'excess demand' which means the quantity demanded is greater than the quantity supplied at the given price so that the company will implement a prime-time charge to increase the price and lower the demand.**

| | ride_prime_time | driver_id | ride_id | ride_distance | ride_duration |
|---|---|---|---|---|---|
| ride_prime_time | 1.00 | -0.56 | -0.56 | -0.56 | -0.56 |
| driver_id | -0.56 | 1.00 | 1.00 | 1.00 | 1.00 |
| ride_id | -0.56 | 1.00 | 1.00 | 1.00 | 1.00 |
| ride_distance | -0.56 | 1.00 | 1.00 | 1.00 | 1.00 |
| ride_duration | -0.56 | 1.00 | 1.00 | 1.00 | 1.00 |

### Data

We use *ride_ids* data frame to calculate the correlation between the PrimeTime multiplier and the drivers/rides

C
u
th
in
*ri*
field which are: 0, 25, 50, 75, 100, 150, 200, 250, 300, 350, 400, and 500

which is -0.56. Also, we have testified that their relationship fits exponential distribution (p-value=0.0).

which is -0.94. Also, we have testified that their relationship fits normal distribution (p-value=1.24e-113).

### ysis

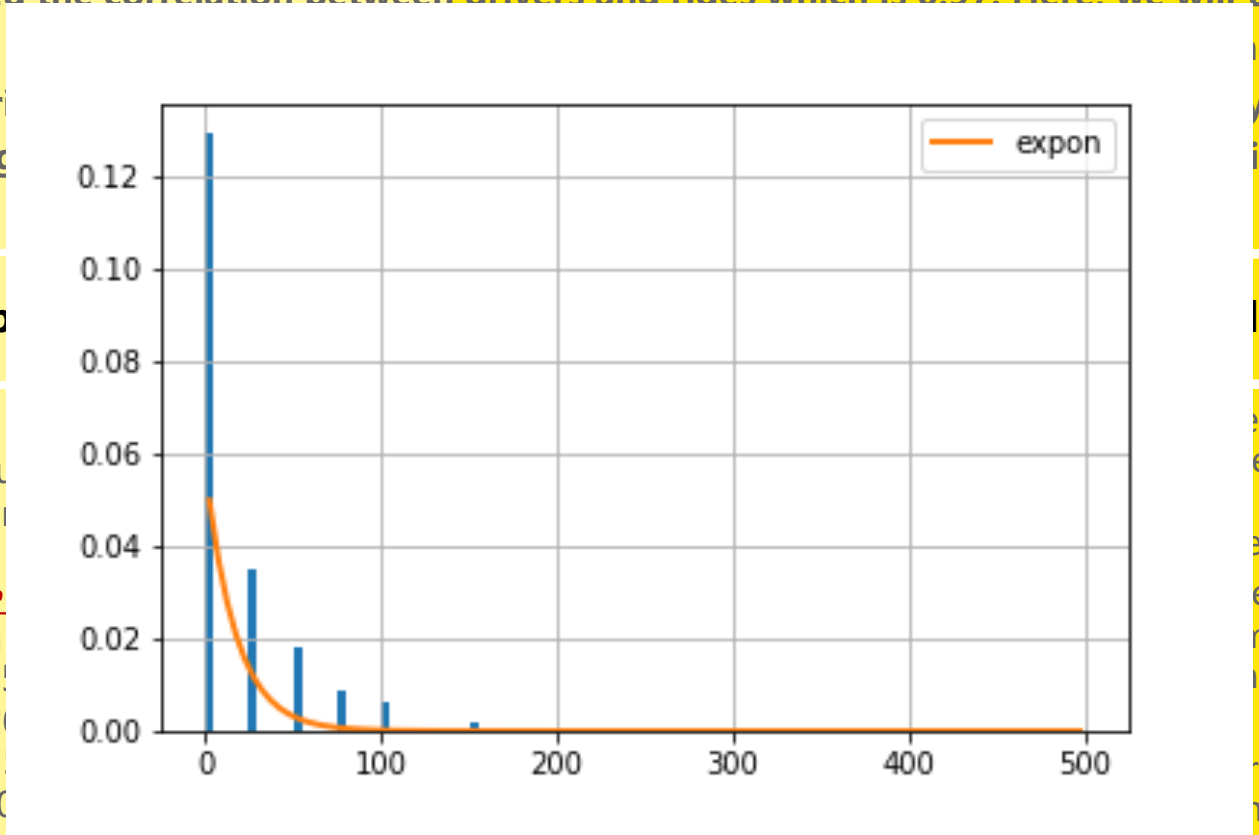
the less the rides
multiplier will be
re for their rides
supply.
the more the drivers, the less the prime-time multiplier will be. When the drivers' number is low, the prime-time price will get more expensive and vice versa.

- **Conclusion**: When the price gets higher, it will attract more drivers to provide more rides and after the supply-demand ratio getting bigger, the price will be pushed lower again.

# Relationship between drivers/rides and prime time

## Introduction

**Previously, we have calculated the correlation between drivers and rides which is 0.97. Here, we will try to find out how the demand for rides impact the e prime-time multiplier is affected by the demand for r y demanded is greater than the quantity supplied at the g increase the price and lower the demand.**

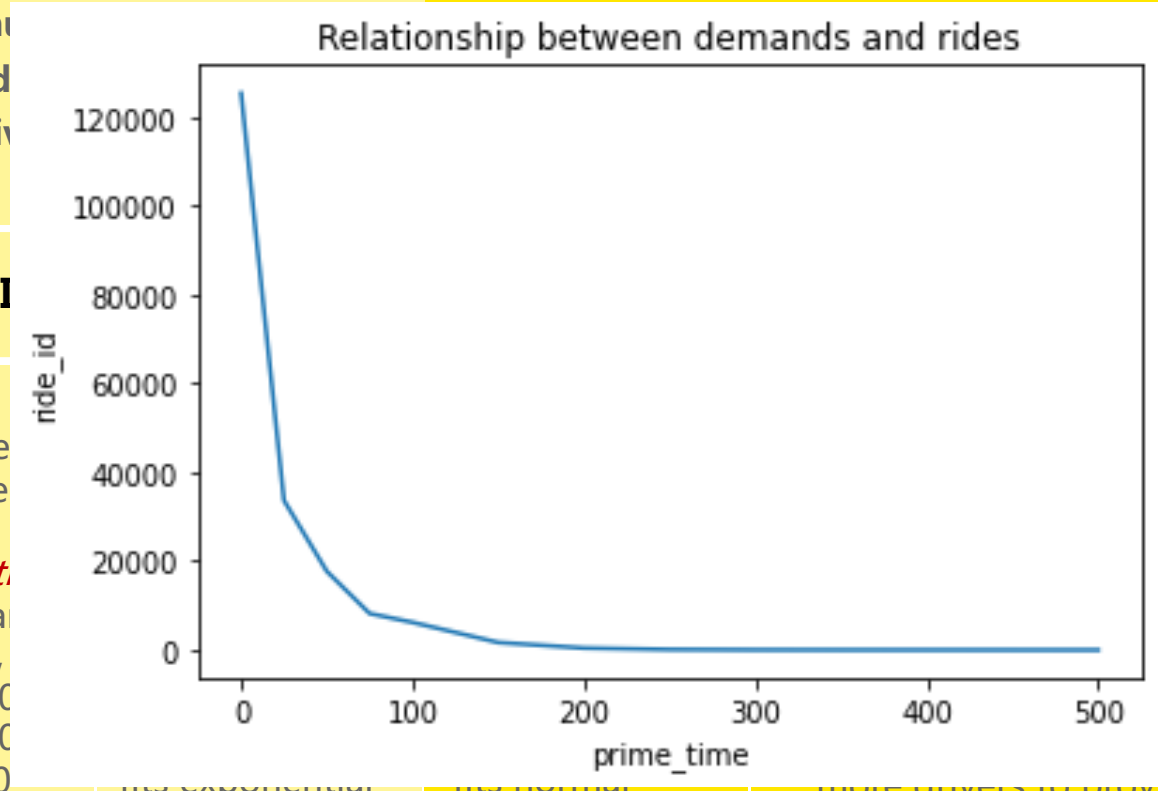



| Data | Step | | lt analysis |
|---|---|---|---|
| We use *ride_ids* data frame to calculate the correlation between the PrimeTime multiplier and the drivers/rides | Check the unique valu that appea in the *ride_prime_* field which 0, 25, 50, 7 100, 150, 2 250, 300, 3 400, and 5 | distribution (p-value=0.0). | distribution (p-value=1.24e-113). | ll that the less the rides e-time multiplier will be pay more for their rides eed the supply.<br>ell that the more the drivers, multiplier will be. When the e prime-time price will get versa.<br>rice gets higher, it will attract more rides and after the supply-demand ratio getting bigger, the price will be pushed lower again. |

# Relationship between drivers/rides and prime time

## Introduction

**Previously, we have calculated the correlation between drivers and rides which is 0.97. Here, we will try to find out how the demand for rides impact the nu at the prime-time multiplier is affected by the demand for rid ntity demanded is greater than the quantity supplied at the giv e to increase the price and lower the demand.**




| Data | Step I | esult analysis |
|---|---|---|
| We use *ride_ids* data frame to calculate the correlation between the PrimeTime multiplier and the drivers/rides | Check the unique value that appeare in the *ride_prime_t* field which a 0, 25, 50, 75, 100, 150, 200 250, 300, 350 400, and 500 | ld tell that the less the rides rime-time multiplier will be will pay more for their rides exceed the supply.<br>uld tell that the more the drivers, me multiplier will be. When the w, the prime-time price will get vice versa.<br>e price gets higher, it will attract ide more rides and after the supply-demand ratio getting bigger, the price will be pushed lower again. |

its exponential distribution (p-value=0.0).

its normal distribution (p-value=1.24e-113).

# Relationship between drivers/rides and prime time

## Introduction

**Previously, we have calculated the correlation between drivers and rides which is 0.97. Here, we will try to find out how the demand for rides impact the number of drivers base on the data we have. We already assume that the prime-time multiplier is affected by the demand for rides. In economics, it's called 'excess demand' which means the quantity demanded is greater than the quantity supplied at the given price so that the company will implement a prime-time charge to increase the price and lower the demand.**

| | ride_prime_time | driver_id | ride_id | ride_distance | ride_duration |
|---|---|---|---|---|---|
| ride_prime_time | 1.00 | -0.94 | -0.94 | -0.94 | -0.94 |
| driver_id | -0.94 | 1.00 | 1.00 | 1.00 | 1.00 |
| ride_id | -0.94 | 1.00 | 1.00 | 1.00 | 1.00 |
| ride_distance | -0.94 | 1.00 | 1.00 | 1.00 | 1.00 |
| ride_duration | -0.94 | 1.00 | 1.00 | 1.00 | 1.00 |

### Data

We use *ride_ids* data frame to calculate the correlation between the PrimeTime multiplier and the drivers/rides

Cl
ur
th
in
*ri*
field which are: 0, 25, 50, 75, 100, 150, 200, 250, 300, 350, 400, and 500

### ysis

which is -0.56. Also, we have testified that their relationship fits exponential distribution (p-value=0.0).

which is -0.94. Also, we have testified that their relationship fits normal distribution (p-value=1.24e-113).

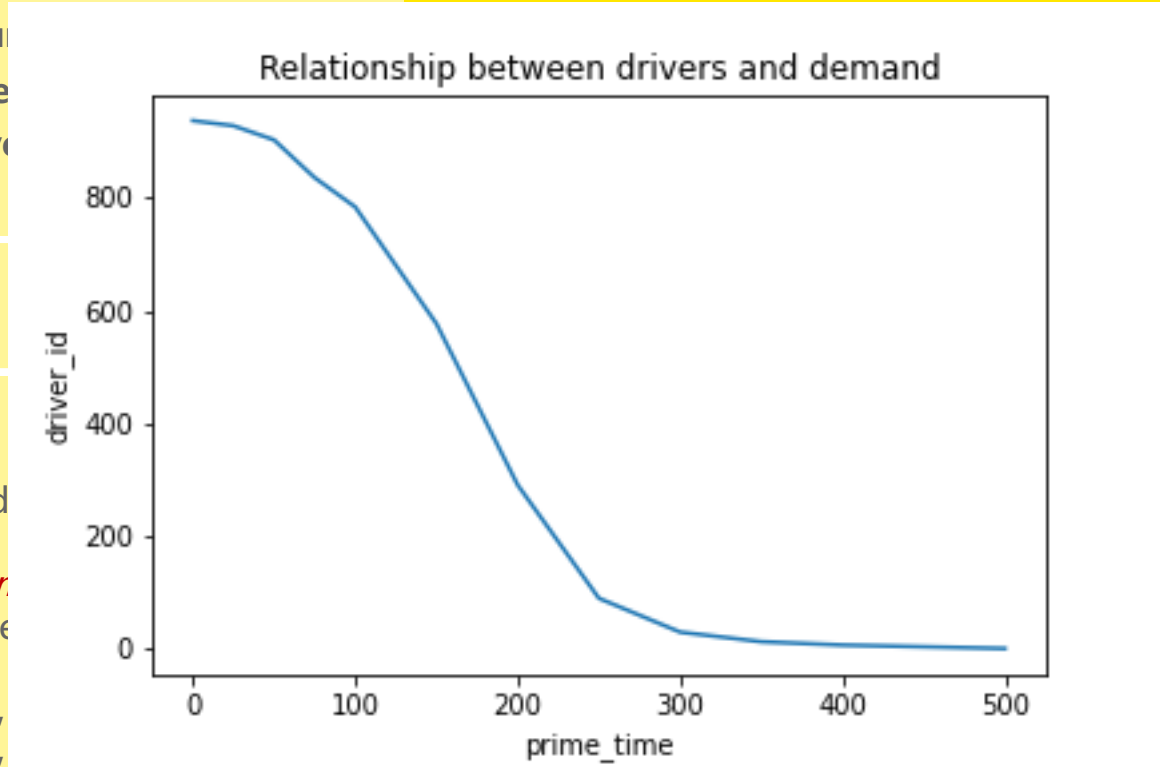
the less the rides
multiplier will be
ore for their rides
supply.
the more the drivers,
the less the prime-time multiplier will be. When the drivers' number is low, the prime-time price will get more expensive and vice versa.

- **Conclusion**: When the price gets higher, it will attract more drivers to provide more rides and after the supply-demand ratio getting bigger, the price will be pushed lower again.

# Relationship between drivers/rides and prime time

## Introduction

**Previously, we have calculated the correlation between drivers and rides which is 0.97. Here, we will try to find out how the demand for rides impact the nu : the prime-time multiplier is affected by the demand for ride tity demanded is greater than the quantity supplied at the giv to increase the price and lower the demand.**

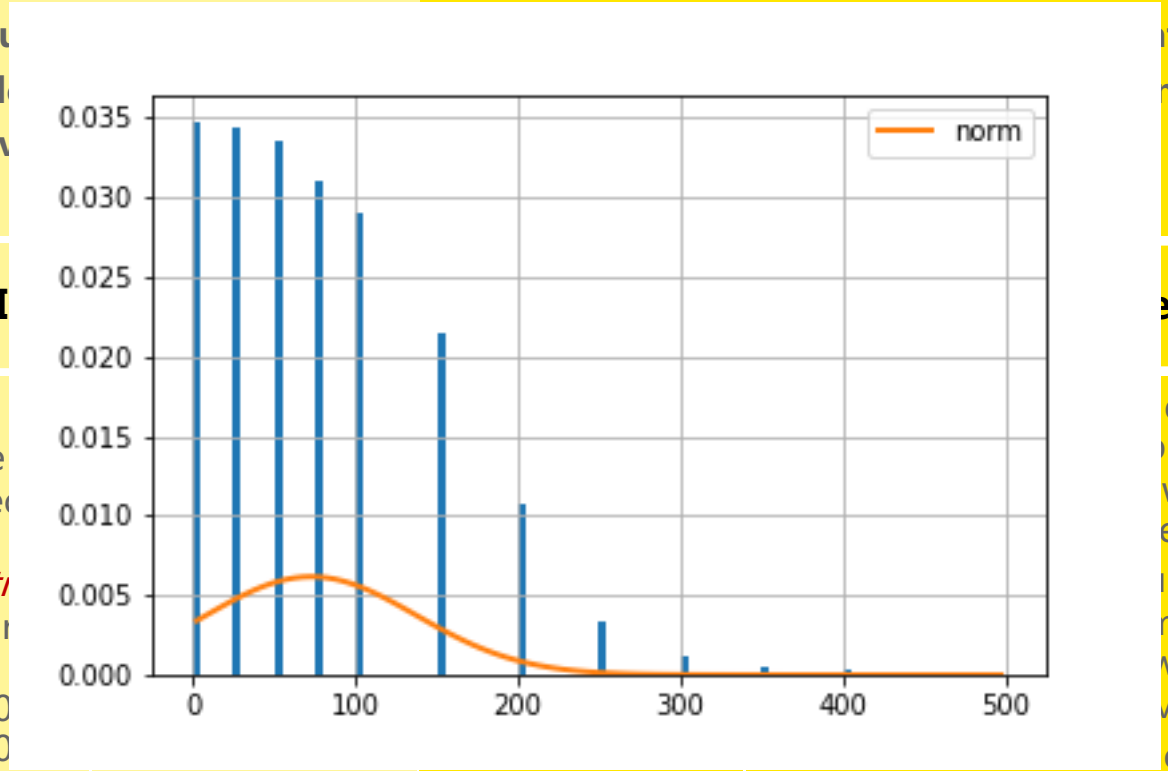



| Data | Step I | | | sult analysis |
|---|---|---|---|---|
| We use *ride_ids* data frame to calculate the correlation between the PrimeTime multiplier and the drivers/rides | Check the unique value that appeared in the *ride_prime_ti* field which are 0, 25, 50, 75, 100, 150, 200, 250, 300, 350, 400, and 500 | fits exponential distribution (p-value=0.0). | fits normal distribution (p-value=1.24e-113). | d tell that the less the rides ime-time multiplier will be will pay more for their rides xceed the supply.<br>d tell that the more the drivers, e multiplier will be. When the , the prime-time price will get ice versa.<br>e price gets higher, it will attract more drivers to provide more rides and after the supply-demand ratio getting bigger, the price will be pushed lower again. |

# Relationship between drivers/rides and prime time

## Introduction

**Previously, we have calculated the correlation between drivers and rides which is 0.97. Here, we will try to find out how the demand for rides impact the nu t the prime-time multiplier is affected by the demand for rid ntity demanded is greater than the quantity supplied at the giv to increase the price and lower the demand.**

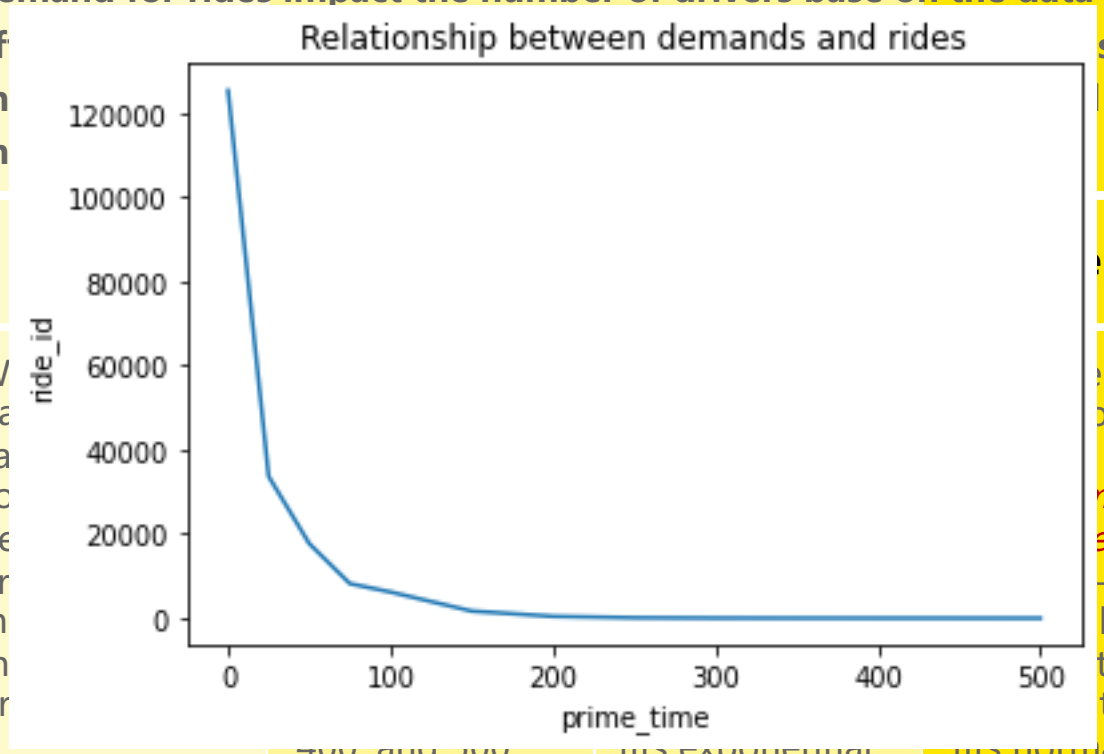



| Data | Step I | | | esult analysis |
|---|---|---|---|---|
| We use *ride_ids* data frame to calculate the correlation between the PrimeTime multiplier and the drivers/rides | Check the unique value that appeare in the *ride_prime_ti* field which ar 0, 25, 50, 75, 100, 150, 200 250, 300, 350 400, and 500 | fits exponential distribution (p-value=0.0). | fits normal distribution (p-value=1.24e-113). | d tell that the less the rides rime-time multiplier will be will pay more for their rides exceed the supply.<br>ld tell that the more the drivers, ne multiplier will be. When the v, the prime-time price will get vice versa.<br>e price gets higher, it will attract more drivers to provide more rides and after the supply-demand ratio getting bigger, the price will be pushed lower again. |

# Relationship between drivers/rides and prime time

## Introduction

**Previously, we have calculated the correlation between drivers and rides which is 0.97. Here, we will try to find out how the demand for rides impact the number of drivers base on the data we have. We already assume that the prime-time multiplier is af s demand' which means the quantity demanded is greater than th implement a prime-time charge to increase the price and lower th**

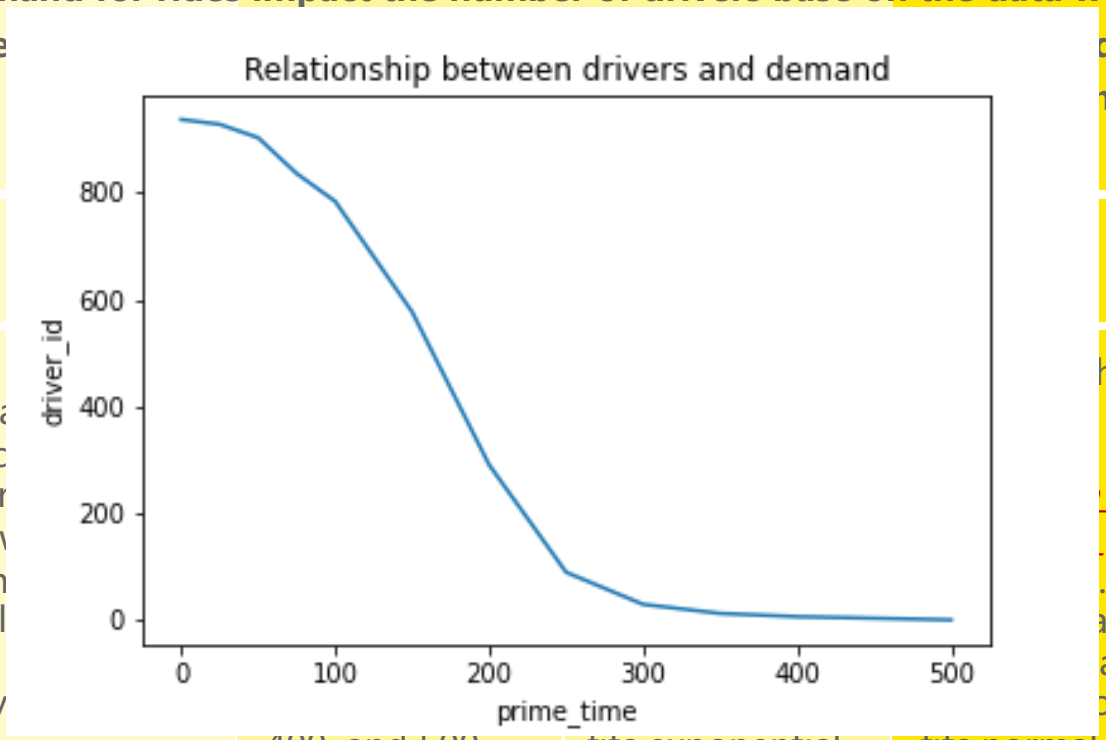



| | | p III | Result analysis |
|---|---|---|---|
| W da ca co be Pr m th dr | 400, and 500 | e the on *ne_time er_id* -0.94. have that tionship | ➢ From **Step II**, we could tell that the less the rides occur, the more the prime-time multiplier will be which means people will pay more for their rides when the demand is exceed the supply.<br>➢ From **Step III**, we could tell that the more the drivers, the less the prime-time multiplier will be. When the drivers' number is low, the prime-time price will get more expensive and vice versa.<br>➢ **Conclusion**: When the price gets higher, it will attract more drivers to provide more rides and after the supply-demand ratio getting bigger, the price will be pushed lower again. |
| | its exponential distribution (p-value=0.0). | its normal distribution (p-value=1.24e-113). | |

# Relationship between drivers/rides and prime time

## Introduction

**Previously, we have calculated the correlation between drivers and rides which is 0.97. Here, we will try to find out how the demand for rides impact the number of drivers base on the data we have. We already assume that the prime-time multiplier is affe… …demand' which means the quantity demanded is greater than the … …mplement a prime-time charge to increase the price and lower the …**




We … data … calc … corr … betw … Prin … mul … the … driv …

400, and 500

fits exponential distribution (p-value=0.0).

**III**

…he …

*…_time …_id*

…94. …ave …at …onship

fits normal distribution (p-value=1.24e-113).

## Result analysis

- From **Step II**, we could tell that the less the rides occur, the more the prime-time multiplier will be which means people will pay more for their rides when the demand is exceed the supply.
- From **Step III**, we could tell that the more the drivers, the less the prime-time multiplier will be. When the drivers' number is low, the prime-time price will get more expensive and vice versa.
- **Conclusion**: When the price gets higher, it will attract more drivers to provide more rides and after the supply-demand ratio getting bigger, the price will be pushed lower again.

# The average projected lifetime of a driver

## Dataset

Because we only have driver onboard date in ***driver_ids*** dataframe, and the onboard month is 03/2016, 04/2016, and 05/2016 in total. We use those driver ids in ***driver_ids*** to apply comparison via driver ids in ***ride_ids_v2***. We check whether those driver ids are appeared in ***ride_ids_v2*** according to the timestamp's month value and for those not appeared, we treat them as churned.

| | 0 | 1 |
|---|---|---|
| driver_id | 002be0ffdc997bd5c50703158b7c2491 | 007f0389f9c7b03ef97098422f902e62 |
| driver_onboard_date | 2016-03-29 00:00:00 | 2016-03-29 00:00:00 |
| leave_april | NaN | NaN |

| | 0 | 1 |
|---|---|---|
| driver_id | 1e9b964b3e3d0289794289579269247a | 7f4350f4a358ac264ccf3b10c4966afc |
| driver_onboard_date | 2016-03-28 00:00:00 | 2016-03-28 00:00:00 |
| leave_may | True | True |

| | 0 | 1 |
|---|---|---|
| driver_id | 1e9b964b3e3d0289794289579269247a | 895c14bfd7d1e2c26aee6938703f32f8 |
| driver_onboard_date | 2016-03-28 00:00:00 | 2016-03-28 00:00:00 |
| leave_june | True | True |

| | 0 | 1 |
|---|---|---|
| driver_id | 02e440f6c209206375833cef02e0cbae | 0eff1404b137a5562642f0f706e59f25 |
| driver_onboard_date | 2016-03-31 00:00:00 | 2016-03-29 00:00:00 |
| leave_april | NaN | NaN |
| leave_may | NaN | True |
| leave_june | True | True |

## Methodology:

- ✓ Find out the drivers that onboard in March and not appear in April / May / June.
- ✓ Merge them into one data frame and calculate the number of drivers that not common for each adjacent month.

**Result:**
March/April: 11/107 = 0.1;
March/May: 19/96 = 0.2;
March/June: 6/77 = 0.08.
**Average projected time of a driver is more than 3 months.**

Where ***leave_month's*** values are 1, 2 and 3 which means leaving within 1 month, 2 months and 3 months.

# 06 Lifetime value(LTV)

# Earnings per drive

## Calculation of earnings per drive

According to the rate card below, we will use this equation to calculate earning for each ride:

**(Base Fare (2) + ride_distance * 0.000621371 * Cost per mile (1.15) + Cost per minute (0.22) * ride_duration / 60 + Service fee (1.75)) * (1 + ride_prime_time) / 100**

Where:

1. **ride_distance * 0.000621371** is to turn meters into miles;
2. **ride_duration / 60** is to turn seconds into minutes;
3. **(1 + ride_prime_time)** is to add prime-time price.

Then, I replace those values that less than 5 and larger than 400 in total field with 5 and 400.

## Overview of the rate card

| | | |
|---|---|---|
| **Base Fare:** $2.00 | **Cost per minute:** $0.22 | **Minimum fare:** $5.00 |
| **Cost per mile:** $1.15 | **Service fee:** $1.75 | **Maximum fare:** $400.00 |

**Before:**

| driver_id | ride_id | ride_distance | ride_duration | ride_prime_time |
|---|---|---|---|---|
| 7bd5c50703158b7c2491 | 006d61cf7446e682f7bc50b0f8a5bea5 | 1811 | 327 | 50 |
| 7bd5c50703158b7c2491 | 01b522c5c3a756fbdb12e95e87507eda | 3362 | 809 | 0 |
| 7bd5c50703158b7c2491 | 029227c4c2971ce69ff2274dc798ef43 | 3282 | 572 | 0 |
| 7bd5c50703158b7c2491 | 034e861343a63ac3c18a9ceb1ce0ac69 | 65283 | 3338 | 25 |
| 7bd5c50703158b7c2491 | 034f2e614a2f9fc7f1c2f77647d1b981 | 4115 | 823 | 100 |
| ... | ... | ... | ... | ... |
| 185ec5e97d59dbcd7a78 | fc717192b3512767269ff5a54b97af05 | 10127 | 1336 | 0 |
| 185ec5e97d59dbcd7a78 | fd6fa5f9265d2cf83936ead663f9e0e7 | 1908 | 445 | 0 |
| 185ec5e97d59dbcd7a78 | fe0857c43025264d337dfe1d8463e503 | 4039 | 875 | 0 |
| 185ec5e97d59dbcd7a78 | ff0db0ca4557bf5b05b4da6f660a1ac1 | 4760 | 777 | 0 |
| 185ec5e97d59dbcd7a78 | ff7dc29693f8c79ff103d350a7b6c157 | 3751 | 889 | 100 |

**After:**

| driver_id | ride_id | ride_distance | ride_duration | ride_prime_time | total |
|---|---|---|---|---|---|
| 3158b7c2491 | 006d61cf7446e682f7bc50b0f8a5bea5 | 1811 | 327 | 50 | 9.364647 |
| 3158b7c2491 | 01b522c5c3a756fbdb12e95e87507eda | 3362 | 809 | 0 | 9.118740 |
| 3158b7c2491 | 029227c4c2971ce69ff2274dc798ef43 | 3282 | 572 | 0 | 8.192574 |
| 3158b7c2491 | 034e861343a63ac3c18a9ceb1ce0ac69 | 65283 | 3338 | 25 | 78.298801 |
| 3158b7c2491 | 034f2e614a2f9fc7f1c2f77647d1b981 | 4115 | 823 | 100 | 19.416299 |
| ... | ... | ... | ... | ... | ... |
| 59dbcd7a78 | fc717192b3512767269ff5a54b97af05 | 10127 | 1336 | 0 | 15.885184 |
| 59dbcd7a78 | fd6fa5f9265d2cf83936ead663f9e0e7 | 1908 | 445 | 0 | 6.745079 |
| 59dbcd7a78 | fe0857c43025264d337dfe1d8463e503 | 4039 | 875 | 0 | 9.844508 |
| 59dbcd7a78 | ff0db0ca4557bf5b05b4da6f660a1ac1 | 4760 | 777 | 0 | 10.000385 |
| 59dbcd7a78 | ff7dc29693f8c79ff103d350a7b6c157 | 3751 | 889 | 100 | 19.380087 |

# Average earnings per driver per month

**1. In order to calculate on a monthly and weekly basis, we merge the ride_ids and ride_ts for time stamp reference and call it ride_ids_v2.**

| | 0 | 1 |
|---|---|---|
| driver_id | 002be0ffdc997bd5c50703158b7c2491 | 002be0ffdc997bd5c50703158b7c2491 |
| ride_id | 006d61cf7446e682f7bc50b0f8a5bea5 | 01b522c5c3a756fbdb12e95e87507eda |
| ride_distance | 1811 | 3362 |
| ride_duration | 327 | 809 |
| ride_prime_time | 50 | 0 |
| event | requested_at | requested_at |
| timestamp | 2016-04-23 02:22:07 | 2016-03-29 19:17:30 |
| total | 9.364647 | 9.11874 |

**2. Group by driver_id of ride_ids_v2 data frame and use the sum method for the rest fields so that we could get the total earnings per driver.**

| | 0 | 1 |
|---|---|---|
| driver_id | 002be0ffdc997bd5c50703158b7c2491 | 007f0389f9c7b03ef97098422f902e62 |
| ride_distance | 1740287 | 117531 |
| ride_duration | 221238 | 20497 |
| ride_prime_time | 5375 | 625 |
| total | 3654.98452 | 332.432167 |

**3. Then we use pd.Grouper to find out the number of months which is 4, so we use the total field divided by 4 and get the average earnings per month.**

| | 0 | 1 |
|---|---|---|
| driver_id | 002be0ffdc997bd5c50703158b7c2491 | 007f0389f9c7b03ef97098422f902e62 |
| ride_distance | 1740287 | 117531 |
| ride_duration | 221238 | 20497 |
| ride_prime_time | 5375 | 625 |
| total | 3654.98452 | 332.432167 |
| avg_week | 261.070323 | 23.745155 |
| avg_month | 913.74613 | 83.108042 |

**Result:**

**Calculate the average value per month of each driver for the company via the following equation and the result is 142.9.**

**Avg_month / number of unique drivers * percentage of earnings for the company per drive (0.2)**

# Calculate churn rate

Month 1

| | 184739 | 192682 |
|---|---|---|
| driver_id | f0df79d10df44f18742682108b17f60a | ff714a67ba8c6a108261cd81e3b77f3a |
| ride_id | 655ccbeb2a62880159e20e986c1cdaeb | e81c8bdc2a6a9056ba49bf67bf97c311 |
| ride_distance | 40812 | 9384 |
| ride_duration | 2112 | 1653 |
| ride_prime_time | 25 | 0 |
| event | requested_at | requested_at |
| timestamp | 2016-03-28 06:37:51 | 2016-03-28 08:41:46 |
| total | 50.821628 | 16.516587 |

Month 3

| | 147552 | 1304 |
|---|---|---|
| driver_id | b7fbf53260861a5a9022b709abe4d584 | 0258e250ca195cc6258cbdc75aecd853 |
| ride_id | 37d45fde6dc8938878927d74e66f7dfc | 608d6814328ca325bfc54acbdc2720e0 |
| ride_distance | 2826 | 3190 |
| ride_duration | 664 | 343 |
| ride_prime_time | 0 | 50 |
| event | requested_at | requested_at |
| timestamp | 2016-05-01 00:00:24 | 2016-05-01 00:00:30 |
| total | 8.20406 | 10.930749 |

Month 2

| | 170458 | 60855 |
|---|---|---|
| driver_id | ddf9fad53fff9a2f2ded181e1144b47f | 4d1e99b879de5fa11e3f5423416f0497 |
| ride_id | a0af1a46129a57162cc0292eef7a5139 | 6dfa817b5ec01a1050328797095824f8 |
| ride_distance | 10473 | 8151 |
| ride_duration | 926 | 1596 |
| ride_prime_time | 0 | 0 |
| event | requested_at | requested_at |
| timestamp | 2016-04-01 00:09:43 | 2016-04-01 00:10:20 |
| total | 14.629095 | 15.426514 |

Month 4

| | 171853 | 2693 |
|---|---|---|
| driver_id | df9ab418e87b21319bdfe389a4552e08 | 2422c0748699c412f9759b2897484067 |
| ride_id | 68b6a11055a3976ce185cec55c1155b9 | 5a753c929fac7c5cf8b8e4049e2532c9 |
| ride_distance | 3964 | 185 |
| ride_duration | 646 | 53 |
| ride_prime_time | 0 | 25 |
| event | requested_at | requested_a |
| timestamp | 2016-06-01 00:00:58 | 2016-06-01 00:01:31 |
| total | 8.951249 | 8.78823 |

1. From ride_ids_v2, split each month into 4 different data frames;
2. Compare churn rate for two adjacent month separately (March/April, April/May, May/June) using the following method:
   - Find the intersection of driver_id in two months (common);
   - (100 – common / unique driver_id * 100)%
3. Calculate the average which is 10.21.

# Lifetime Value (LTV)

**Previously, we calculate the average value per month of each driver for the company & churn rate.**

**1. Average lifetime**

- Monthly churn rate at 10.21% puts the average lifetime at 10 months. (1 / 10.21%)

**2. Average value**

- Average value per month of each driver for the company is 143.

**3. Lifetime value**

- We could get lifetime value via 10 * 143, which is 1430, meaning the LTV of each driver is $1430.

# 07 Feature Selection

# Feature Selection for lifetime value

**Purpose**

Find out the main factors that affect a driver's lifetime value using feature selection method (function: f_classif).

## Calculation

| features | score |
|---|---|
| ride_distance | 665446.682397 |
| ride_duration | 379903.181438 |
| ride_prime_time | 23446.939912 |
| driver_onboard_date | 46.512304 |
| leave_month | 44.409173 |

| | ride_distance | ride_duration | ride_prime_time | total | leave_month | driver_onboard_date |
|---|---|---|---|---|---|---|
| ride_distance | 1.00 | 0.79 | -0.05 | 0.88 | 0.01 | 0.00 |
| ride_duration | 0.79 | 1.00 | 0.01 | 0.81 | 0.01 | 0.01 |
| ride_prime_time | -0.05 | 0.01 | 1.00 | 0.33 | 0.02 | 0.03 |
| total | 0.88 | 0.81 | 0.33 | 1.00 | 0.02 | 0.02 |
| leave_month | 0.01 | 0.01 | 0.02 | 0.02 | 1.00 | 0.13 |
| driver_onboard_date | 0.00 | 0.01 | 0.03 | 0.02 | 0.13 | 1.00 |

## Analysis

The result shows that among all the features, ride distance affect a driver's lifetime value most and second important feature is ride duration and then is ride prime time's multiplier which make sense because the lifetime value of drivers is generated by those three values. Other than that, driver's onboard date and driver's leave month are also slightly impact the lifetime value, the more time a driver spend to drive, the more value he/she will generate. As all the correlations between variables and target variable are positive, they all positively impact the lifetime value.

# Feature Selection for churn rate

## Purpose

Find out the main factors that affect a driver's churn rate using feature selection method (function: f_regression) and segment the driver population to identify driving behavior that may lead to churn.

## Calculation

| features | score |
|---|---|
| driver_onboard_date | 1247.879805 |
| ride_prime_time | 19.915761 |
| total | 17.996697 |
| ride_distance | 11.801586 |
| ride_duration | 7.992708 |

| | ride_distance | ride_duration | ride_prime_time | total | leave_month | driver_onboard_date |
|---|---|---|---|---|---|---|
| ride_distance | 1.00 | 0.79 | -0.05 | 0.88 | 0.01 | 0.00 |
| ride_duration | 0.79 | 1.00 | 0.01 | 0.81 | 0.01 | 0.01 |
| ride_prime_time | -0.05 | 0.01 | 1.00 | 0.33 | 0.02 | 0.03 |
| total | 0.88 | 0.81 | 0.33 | 1.00 | 0.02 | 0.02 |
| leave_month | 0.01 | 0.01 | 0.02 | 0.02 | 1.00 | 0.13 |
| driver_onboard_date | 0.00 | 0.01 | 0.03 | 0.02 | 0.13 | 1.00 |

## Analysis

The result shows that among all the features, driver's onboard date effect the drivers' churn rate most and second is the ride prime time's multiplier, then total value that drivers make. As all the correlations between variables and target variable are positive, they all positively impact the lifetime value.

# 08 Recommendation & Conclusion

# Recommendation & Conclusion

According to the existing data set's variables, we learned that the average lifetime of a driver is about 10 months; average lifetime value is about $1430; the churn rate per month is about 10.21%. Besides, we learned the relationship between drivers and rides, rides' number and prime time multiplier, drivers' number and prime time multiplier are positively related to each other. Besides the features that we evaluated previously, included in the data sets we used, there are many other components that might affect the driving business, like drivers' license, accidents, driving experience, attitude to customers etc.

### Driving experience

**Motor vehicle record contains the driving history of the employee, any violations, suspensions, and accidents which will help the employer choose qualified sate drivers as well as build reputation.**

### Deregulate fares

**Almost all driving company like uber set the fares for all rides, including the drop rate and per kilo costs. It might be useful if individual taxi drivers are allowed to set their own rates if the ride is pre-booked.**

### Cashless transactions

**Fare evasion is a constant threat for taxi drivers, which leads to drivers profiling passengers, and passengers increasingly worry about becoming victims of financial fraud if they use a debit or credit card.**


THANK YOU!
AI LAUNCH LAB