problem 1 : Data Analysis

aim : to increase the accuracy of predicted prices for our ride hailing platform

date_range for the dataset: 2020-02-14 to 2020-03-13

total orders during this period : **4147**

note :

1. total unique order count is 4166, of which 19 were dropped because we do not have upfront and metered prices for these
2. there will remain cases where we have metered price but no upfront price, we have assumed that for these cases, upfront price is the same as metered price, and that prediction was accurate

In order to improve the prediction accuracy, first, we need to understand what the existing prediction accuracy is

a price **prediction is considered inaccurate** if the percentage change between upfront and metered price is **greater than 20%**

applying this logic to the dataset in question,

bad prediction orders : 1352
total orders : 4147
percentage of incorrectly predicted orders : 32.6

TODO : reduce the percentage of inaccurately predicted orders to significantly less than 32 percent of all orders placed

based on the information that is available, a few things that likely have a relationship with prediction accuracy are :

GPS Confidence

looking into the relationship between prediction logic and gps confidence, we get the following results -

orders with good gps confidence and correct prediction : 54%
total orders with good gps and wrong prediction : 28%
total orders with bad gps and correct prediction : 13%
total orders with bad gps and correct prediction : 5%

conclusion :
even for orders with good gps score, which is 82 percent of all orders, prediction accuracy is ~66%

it does not seem like working on our navigations systems will have a direct impact on improving price predictions

Entered By

the address to the destination of the order can either be entered by the driver or the client. As the apps that will be used by both these user cohorts are different, it is possible that the prediction is more accurate for one app

entered by client and correct prediction : 64%
entered by client and wrong prediction : 32%
entered by driver and correct prediction : 4%
entered by driver and wrong prediction : 1%

conclusion :
96 percent orders have addresses entered by users, which is inaccurate 1 in 3 times, i.e, prediction accuracy is ~33 percent

of the 4 percent of orders where destination is entered by the user, prediction is off ~ 1 in 4 times, or prediction accuracy is 75 percent.
but as this is applicable to very few orders (only 4 percent), it is not seem reasonable to draw such conclusions

Destination Changes

it is possible that the prediction price value is off when there are destination changes.

destination not changed and correct prediction : 61%
destination not changed and wrong prediction : 31%
changed and correct prediction : 6%
destination changed and correct prediction : 2%

92 percent of cases, destination is only set once, and accuracy for these cases is roughly 60%

similarly, the cases where destination change takes place, the accuracy is ~75% but this is for a very limited set of orders.

Prediction Price Type

there are 4 types of prediction_type variables in our system.
pre ride prediction : upfront, prediction
prediction after rider changed the destination : upfront_destination_changed

UPFRONT and correct prediction : 39%
UPFRONT and wrong prediction : 33%
UPFRONT DEST changed and correct prediction : 4%
UPFRONT DEST changed and wrong prediction : 0%
UPFRONT WAYPOINT changed and correct prediction : 0%
UPFRONT WAYPOINT and wrong prediction : 0%
PREDICTION and correct prediction : 24% (order-count) : 998
PREDICTION and wrong prediction : 0%

we notice that when prediction variable is `upfront`, the accuracy of the prediction is ~50%
but for the 998 orders where prediction variable was `prediction`, accuracy was 100%

Conclusion
---

for most orders, price prediction happens upfront, i.e, dest_change_number = 92% in the given dataset. For cases where there are no destination changes, price prediction needs to happen only once, which is before the ride
type:

Upfront : prediction correctness accuracy is 54% type: Prediction : accuracy is 100%
based on the information at hand, we can assume that price_prediction_type = prediction is more accurate

if all of these orderes, price prediction had been of type prediction, accuracy would be : (1615+1352+988)/4147 = 95%
so, we would have gotten down the predicition inaccuracy from 32 percent to ~5% percent, which is significantly better.

note :
1. this is based on the assumption that there is a different price prediction logic that is used for both of these types
2. there is also the fact that its unlikely that the prediction accuracy is actually 100 percent, and we should look at more data to draw a more accurate precision figure

App Type

we have different types of user app builds, CA and CI

	user_app_type	user_app_version	total_bad_prediction_orders	total_orders	bad_prediction_percentage
0	CA	4	34	116	29.310345
1	CA	5	679	2210	30.723982
2	CI	3	14	51	27.450980
3	CI	4	625	1770	35.310734

across these versions, it looks like CA app types prediction accuracy is better than CI by 5 percentage points. Might be something worth looking into

Fraud Contribution to Bad Score

fraud orders : 41
fraud orders with bad prediction : 7
bad prediction attributable to fraud orders percentage : 17%

does not look like orders with nonZero fraud scores have bad prediction accuracy