asSQL

/\*Q1

For request times between 12/1/2013 10:00:00 PST & 12/8/2013 17:00:00 PST, how

many completed trips (Hint: look at the trips.status column) were requested on iphones

in City #5? on android phones?

\*/

select Count(id) as CompletedTrips #count(id) = complete trips

from trips

where status='completed' and (request\_device='iphone' OR request\_device='android')

and city\_id=5

and request\_at >= '2013-12-01 10:00:00'

AND request\_at <= '2013-12-08 17:00:00'

/\*Q2

In City #8, how many unique, currently unbanned clients requested a trip in October

2013 that was eventually completed?

\*/

select count(distinct client\_id) as TotalClients

from

trips

inner join Users U on t.client\_id=u.UserID and u.role='client' and u.banned=0

where city\_id=8

and year(request\_at)=2013 and month(request\_at)=10

and status='completed'

/\*Q2- Part B

Of these, how many trips did each client take?

\*/

select distinct u.UserID as ClientID, U.firstname, U.lastname, Count(t.id) as TotalTrips

from

trips t

inner join Users U on t.client\_id=u.UserID and u.role='client' and u.banned=0

where city\_id=8

and year(request\_at)=2013 and month(request\_at)=10

and status='completed'

group by u.UserID , U.firstname, U.lastname

Part A  
1.     Accuracy of automatically-populated earnings on tax forms

2.     If calls are monitored for reports of inaccuracy by drivers, the # of calls/reports would be a good metric to measure the success of the accuracy of the forms

3.     Develop a tool (e.g., python pipeline similar to Part B) that compares the automatically populated tax forms’ data to the underlying source where the data is derived from (payroll system, etc.)

Part B

Initial model produced a low R^2 ~40% when I ran a linear regression on

Gross\_Bookings ~ Month + C(Vendor\_Name) + C(Country) + C(Product)  
  
This is very bad for a final model, but is fine for an exploratory model that could be tweaked especially since most of the labor for this project included cleaning the data, building the pipeline, and loading the data into the appropriate data structures. This means one/two things. That some of the data points are heavily skewing the results. This is the case with some of the larger values once products and vendors are chosen on introspection. Some products such as Product 1 and Product 16 also seem to be much more difficult in modeling. Linear model either needs to be tweaked using non-linear parameters on some products. This is also expected of a model whose power largely relies on categorical data.

One assumption that I made with cab sharing service, is that users will use the product no matter what the fees are. That is to say revenue is not very dependent on the fees cab sharing service may have to pay out to drivers on its ride sharing service, and this shouldn’t affect the end user wanting to use the app more or less.

After my initial model was poor I chose to filtered my data on product first on the product. I noticed many points where the product in a vendor and country was only existent for a month. That is not enough data to build out a model on. For instance in finance finding seasonality in just a year and half of data is difficult. Much more appropriate would be seasonality is in months and quarters. More data would be needed. So we must disclude or penalize for such months.   
  
I then discluded country altogether and just aggregated on product. To predict the latest date (first of the month for April if I am predicting May), I built a regression on that product’s gross bookings, with enough data, and on most product lines achieves an R^2 of 85% or greater.  
  
Since the fee structure is not fixed and varies widely, one can better use the fee structure (its mean or study its variance, or set it accordingly) as an input into a model to predict the final gross expenses.