Optimizing NYC Taxi Gratuities with ML

Modeling NYC Taxi Data in the Cloud with BigQuery

Lora Johns

DS4A

October 23, 2019

Outline

Can we predict tips for cab drivers in NYC?

- Public transportation is down, and ride-sharing usage is up. (Pew Research)
- Why care about tips?
  - Identify high value times and places
  - Help MTA understand traffic patterns
- Insights could help optimize driver earnings and identify areas that need more bus or metro access.

Research questions

- Patterns in taxi usage over time
  - When are taxis most heavily used?
  - Which geographic zones rely most heavily on taxis?
- What factors are correlated with high tips?
  - What are the strongest predictors of tips?
  - What other data contribute, e.g., weather or demographics?

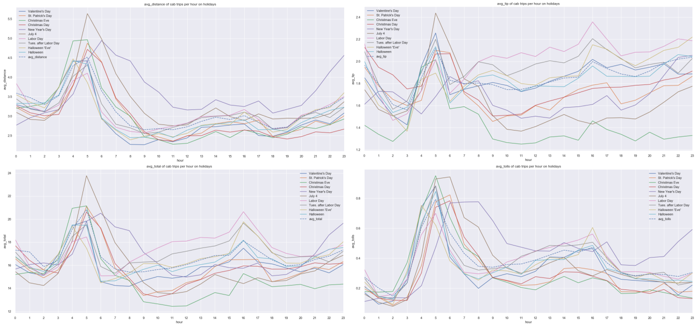
NYC Yellow Cab Data

- Taxi and Limousine Commission data from 2018
  - The TLC released public taxi data from 2009 to present.
  - Available free to access on Google BigQuery.
- What's in the ride data?
  - For 2018, the database contains 112,234,626 records of Yellow Cab rides
  - Records include pick-up and drop-off dates /times, locations, trip distances, itemized fares, rate types, payment types, and driver-reported passenger counts

Data Preparation

- Data Cleaning
  - We must eliminate erroneous records before modeling—some data were poorly collected or transmitted by the sensors.
  - We find and remove observations with negative ride durations, negative tip, fare, or distance, duplication, zero passengers, etc.
  - Make note of outliers, for later modeling
- Data Exploration
  - We queried the public data with BigQuery's native SQL to find the distributions and correlations of the variables of interest.
  - Some notable outliers seemed like errors, but others raised interesting questions
    - Why is there a series of $10,000 trips all with plausibly long distances?
    - What's the pattern behind the 3 million trips between 0 and 1 mile?

The average day vs. selected holidays in 2018



BigQuery for Machine Learning

- Why BigQuery?
  - BigQuery is a serverless, scalable, and democratic cloud data warehouse.
  - It hosts many public datasets that users can join to their uploaded data.
  - Its data structure for nested records and distributed, tree-based query engine mean that it can execute ad-hoc SQL faster than if the data were stored in a more usual format.
- Training ML models in the cloud
  - We queried the public data and trained a model natively to take advantage of Dremel and the wealth of public data.
  - BigQuery's native SQL integrates with Python and R to visualize and additionally analyze queries.

Feature Engineering

- Time and geographic data
  - Extracting the hour, day, and month allows us to granularly analyze trip patterns over time by neighborhood and borough.
  - We can check whether a given (lat, long) is inside a taxi zone polygon by solving a system of linear equations.
- Additional features
  - holidays
  - days of week
  - rush hour
  - overnight trip
  - weekend
  - airport trip

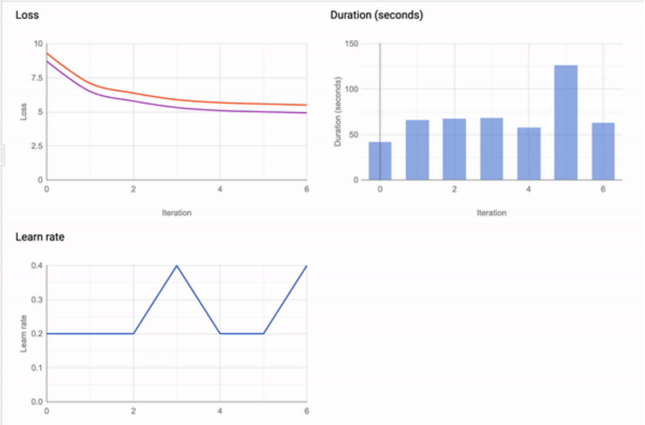
Basic linear regression

- Base model
  - We trained a linear regression model using L1 regularization and batch gradient descent.
  - Using a hash function on unique row timestamps, we pseudorandomly and reproducibly split the data into train-test and evaluation sets.
- Training and evaluating
  - We trained the model in the cloud using ML.CREATE, with 20% of data for testing.
  - The base model explained 0.63 percent of the variance in the outcome (but we wouldn't expect linear regression to be the optimal model for tips).
  - Using ML.EVALUATE on the holdout data, we see that our R^2 value increased by 0.3, indicating that we have avoided overfitting.
  - With ML.PREDICT, we can see the model's actual numerical predictions.

A sample query to evaluate a model

```
SELECT tip_amount, predicted_tip_amount
FROM ML.PREDICT(MODEL `nyc-transit-256016.nyc_taxi.tips_model_L1`, (
       SELECT
         --datetime info
         EXTRACT(MONTH FROM pickup_datetime) AS pickup_month,
         FORMAT_DATE('%A',DATE(pickup_datetime)) as weekday_name,
         EXTRACT(DAY FROM pickup_datetime) AS p_day,
         EXTRACT(HOUR FROM pickup_datetime) AS p_hour_of_day,
         EXTRACT(DAY FROM dropoff_datetime) AS d_day,
         EXTRACT(HOUR FROM dropoff_datetime) AS d_hour_of_day,

         --general ride info
         passenger_count,
         trip_distance,

         --dollar info
         fare_amount,
         mta_tax,
         tolls_amount,

         --categorical variables
         payment_type,
         is_weekend,
         is_airport,
         is_peak,

         --geographical info
         pickup_location_id,
         dropoff_location_id,
         tip_amount

       FROM
         `nyc-transit-256016.nyc_taxi._model_data_table` -- the table I created
       WHERE
         trip_distance > 0 AND fare_amount BETWEEN 0.01 AND 3000.0
         AND DATETIME_DIFF(dropoff_datetime, pickup_datetime, HOUR) > 0 -- Filters out all the stuff we
don't want to train on
         AND passenger_count > 0
         AND tip_amount >= 0
         AND MOD(ABS(FARM_FINGERPRINT(CAST(pickup_datetime AS STRING))),10) >= 8
       )
      )
```

Batch gradient descent



Top tips

- Queens tops the tip list.
  1. Westerleigh
  2. Newark Airport
  3. Saint Michaels Cemetery/Woodside
  4. Astoria Park
  5. Jamaica Bay
  6. Flushing Meadows-Corona Park
  7. Randalls Island
  8. LaGuardia Airport
  9. Rikers Island
  10. Baisley Park

Low rides

- Staten Island is the most negatively correlated borough.

1. Arden Heights
2. Stapleton
3. Bloomfield/Emerson Hill
4. Far Rockaway
5. Charleston/Tottenville
6. Port Richmond
7. New Dorp/Midland Beach
8. Saint George/New Brighton
9. Rosedale
10. Mariners Harbor

Interesting findings

- Thursday has the highest correlation with tips. Saturday has the lowest.
- The feature most strongly correlated with tips was the engineered airport variable.
- Toll amount, trip distance, fare amount, and hour of the day were the next most correlated.

Next steps

- The target and the cardinality of the data mean that a multinomial regression or other model will likely perform better.
- We will visualize the ride map and engineer features such as distance from metro stops.
- Examining the model's coefficients surfaced interesting patterns in the data that will improve the input data.
- BigQuery allows for uploading TensorFlow models, which may be a fruitful avenue to pursue.

Thank you!