Introduction to Big Data

Taxi trip price prediction

Prepared by:
Alexandra Vabnits
Bulat Akhmatov
Nursultan Abdullaev
Ruslan Izmailov

Project Goal and Vision

This project focuses on building a predictive ML model to **estimate the total fare amount** for yellow taxi rides in New York City.

Our goal is to accurately forecast the total amount charged to passengers based on various ride-related features available.

This may be useful for passengers, aggregation taxi platforms and as a competition analysis for business owners.

Dataset characteristics

Features:

- VendorID	INTEGER
- tpep_pickup_datetime	TIMESTAMP
- tpep_dropoff_datetime	TIMESTAMP
- passenger_count	INTEGER
- trip_distance	FLOAT
- pickup_longitude	FLOAT
- pickup_latitude	FLOAT
- RatecodeID	INTEGER
- store_and_fwd_flag	CHAR(1)
- dropoff_longitude	FLOAT
- dropoff_latitude	FLOAT
- payment_type	INTEGER
- fare_amount	FLOAT
- extra	FLOAT
- mta_tax	FLOAT
- tip_amount	FLOAT
- tolls_amount	FLOAT
- improvement_surcharge	FLOAT
- **total_amount**	FLOAT

Parameters:

- 12 210 952 rows
- 1.91 Gb

VendorID	Pickup Time	Dropoff Time	Passengers	Distance (mi)
2	"2016-03-01 00:00:00"	"2016-03-01 00:07:55"	1	2.50
1	"2016-03-01 00:00:00"	"2016-03-01 00:11:06"	1	2.90

Pickup (lon, lat)	Dropoff lon	RateCodeID	Flag	Payment Type
"−73.9767, 40.7652"	−74.0043	1	"N"	1
"−73.9835, 40.7679"	−74.0059	1	"N"	1

Data Analysis - 1/7

To understand our data, we started with identifying the most useful features: the ones that have a high correlation with the target.

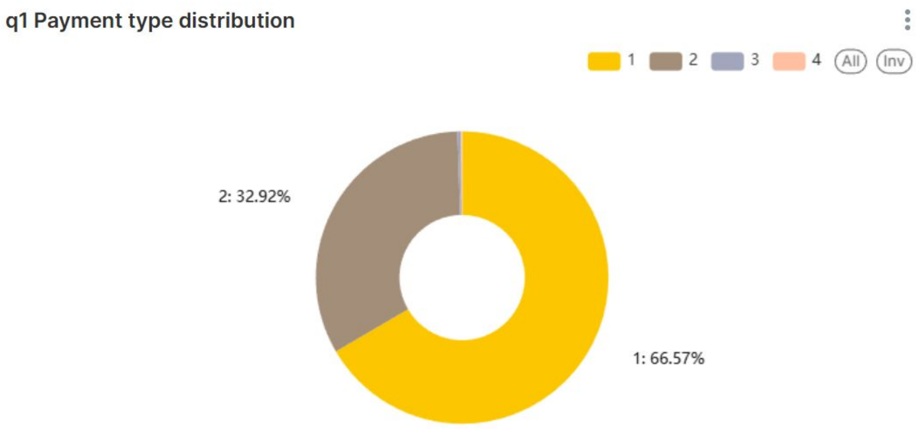
Metric	Total (Sum)
corr_total_amount_passenger_count	265µ
corr_total_amount_trip_distance	0.4172
corr_total_amount_pickup_longitude	0
corr_total_amount_pickup_latitude	0
corr_total_amount_fare_amount	0.9996
corr_total_amount_extra	0.0631
corr_total_amount_mta_tax	-0.0182
corr_total_amount_tip_amount	0.0692
corr_total_amount_tolls_amount	0.0593
corr_total_amount_improvement_surcharge	-0.0046

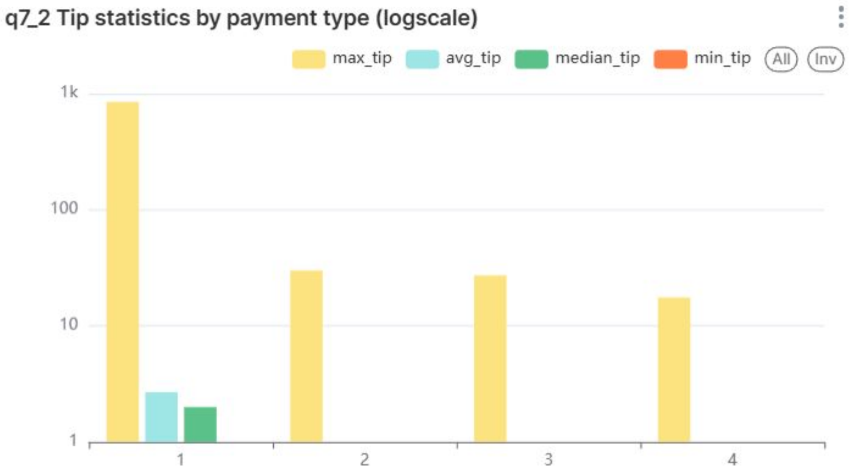
Data Analysis - 2/7

We decided to explore payment types

A numeric code signifying how the passenger paid for the trip.

1. Credit card
2. Cash
3. No charge
4. Dispute



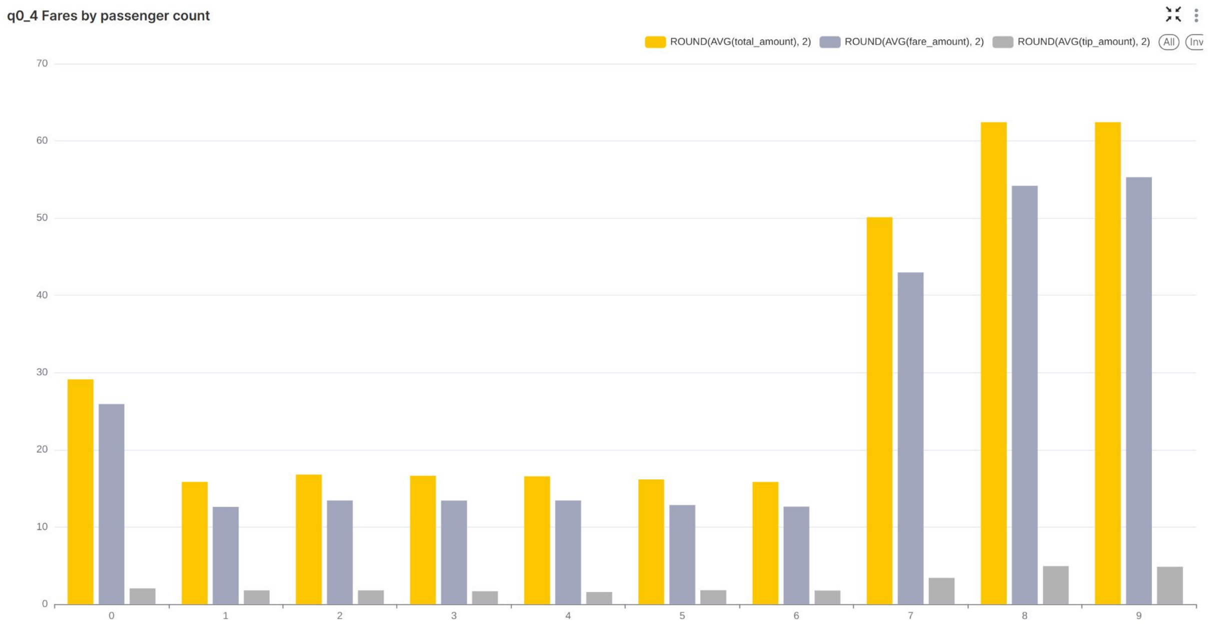


Cash tips are not included into tip_amount

Data Analysis - 3/7

There is an obvious difference for different passenger counts. Why is the correlation weak?

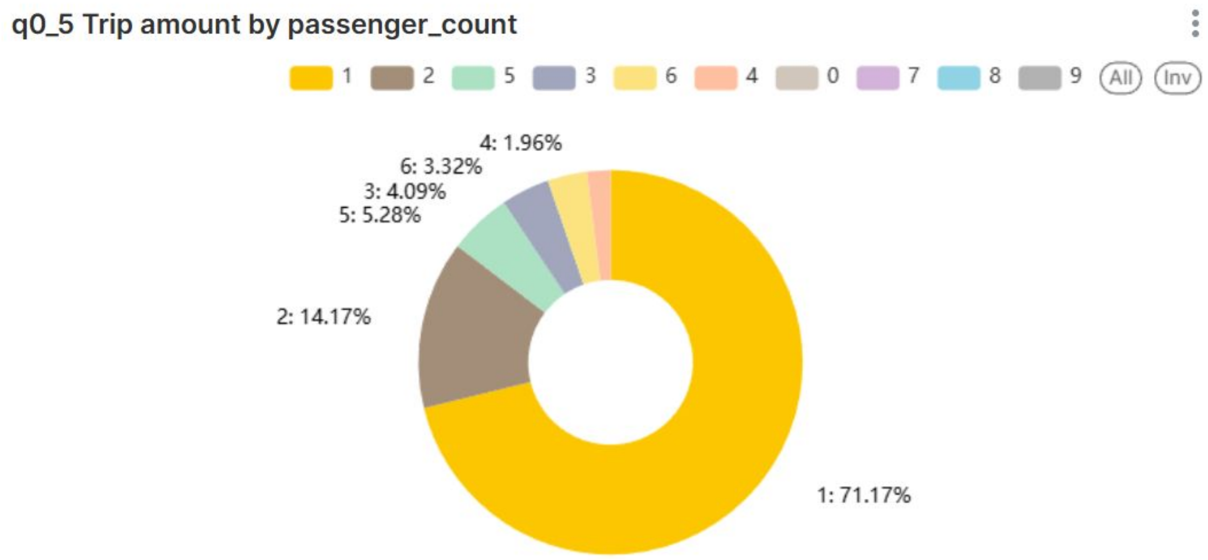
0 and 7-9 passenger counts are very rare



Data Analysis - 3/7

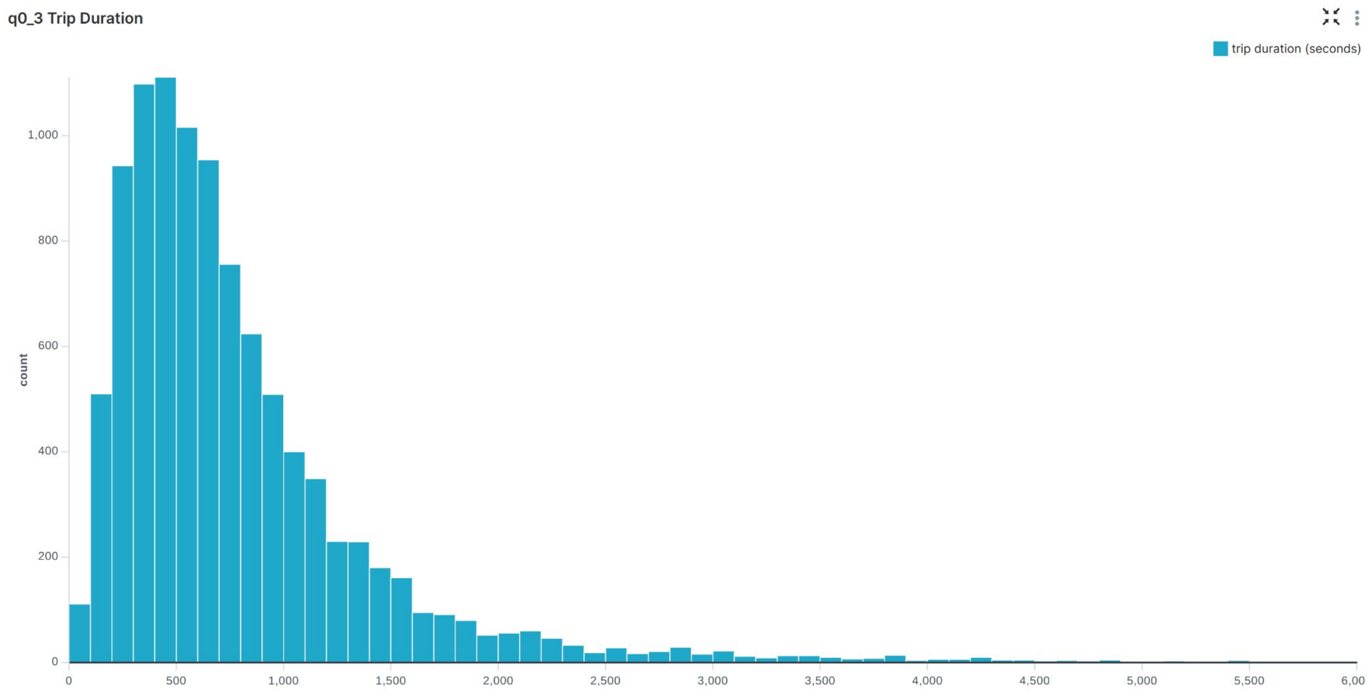
There is an obvious difference for different passenger counts. Why is the correlation weak?

0 and 7-9 passenger counts are very rare



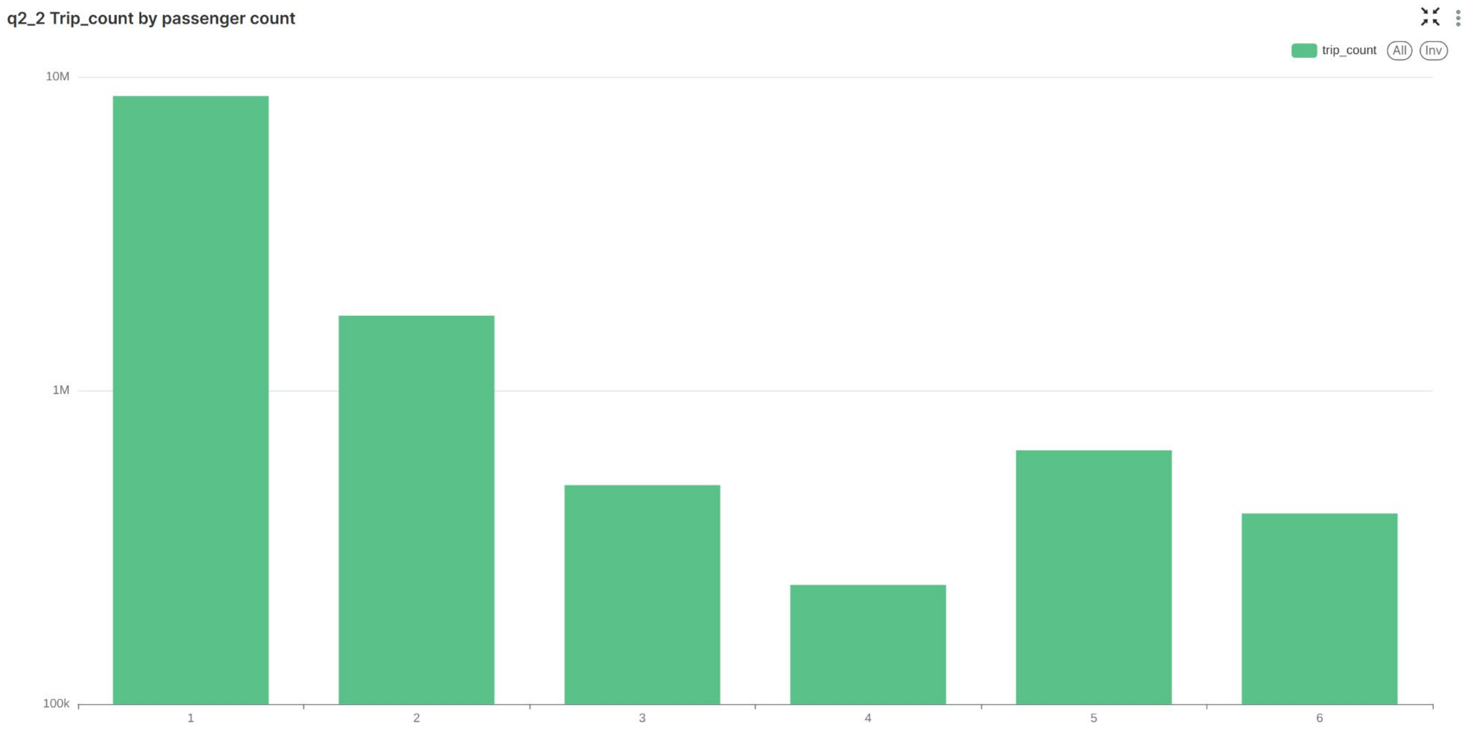
Data Analysis - 4/7

Trip duration (custom feature) histogram



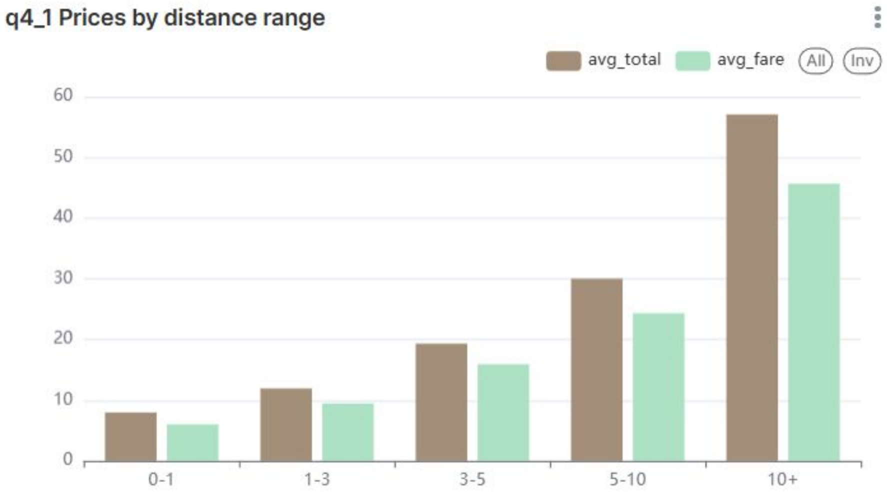
Data Analysis - 5/7

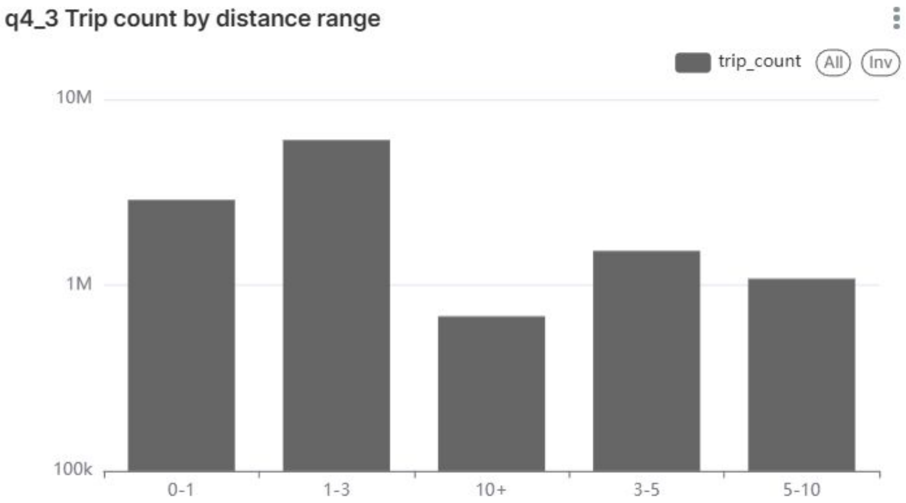
Trip count by passenger count



Data Analysis - 6/7

Prices and trip count by distance range in miles





Data Analysis - 7/7

Some features have unnatural negative values

stat	fare_amount	extra	mta_tax	tip_amount	tolls_amount	improvement_surcharge	total_amount
count	123k	123k	123k	123k	123k	123k	123k
median	9.5	0	0.5	1.35	0	0.3	11.8
max	400	4.5	0.65	120.05	100.99	0.3	520.35
mean	12.76	0.3445	0.4976	1.79	0.3161	0.2997	16
min	-68	-1	-0.5	0	0	-0.3	-68.8
std	11.04	0.5087	0.0377	2.49	1.42	0.0124	13.63

Analysis results

- Pretty natural data with logically explainable correlations
- There are unnatural outliers sometimes
- Coordinates are not important (linearly)
- Trip distance is very important

Project stages - 1/4 | Data Collection & Ingestion

Work done:

1. Prepared git repository
2. Cluster **workspace prepared**
3. Dataset downloaded
4. Created **postgres database** with data loaded in it
5. Data imported to **HDFS**
6. All tasks automated

Project stages - 2/4 | Data Preparation & EDA

Work done:

- **Hive database** is built
    1. Hive tables are prepared
    2. Partitioning added
    3. All tasks automated

- **EDA** is performed
    1. Correlation analysis
    2. Data distribution charts
    3. All charts saved in superset

Project stages - 3/4 | Modeling

Work done:

1. Read data from hive tables
2. Built and fit a **feature extraction pipeline**
3. Build initial **models**
4. Performed **hyperparameter tuning** for 2 models
5. **Predicted samples** with baseline and tuned models
6. **Evaluated** the performance of baseline and tuned models
7. **Saved results** and models to hdfs and local file system
8. All tasks automated

Goal: price prediction
Task: regression

Preprocessing of data

Preprocessing steps:

1. **Timestamp Conversion:** Convert Unix ms to timestamp (pickup & dropoff)
2. **Time Features:** Extract hour and month from timestamps
3. **Coordinate features:** Normalize longitude & latitude
4. **Cyclical Encoding:** Encode time and coordinate features using sine & cosine
5. **Feature Selection:** Select relevant columns, rename target to label
6. **Vectorization:** Assemble features into features_raw
7. **Scaling:** Standardize features with StandardScaler → features

Selected features + total_amount as target:

```
inputCols=[
    'vendorid',
    'passenger_count', 'trip_distance',
    'pickup_lon_sin', 'pickup_lon_cos',
    'pickup_lat_sin', 'pickup_lat_cos',
    'dropoff_lon_sin', 'dropoff_lon_cos',
    'dropoff_lat_sin', 'dropoff_lat_cos',
    'pickup_hour_sin', 'pickup_hour_cos',
    'pickup_month_sin', 'pickup_month_cos',
    'dropoff_hour_sin', 'dropoff_hour_cos',
    'dropoff_month_sin', 'dropoff_month_cos'
```

Example of preprocessed data:


```
{"features":{"type":1,"values":[-1.0630742931954804,-0
.5027240155344938,-6.316393234651101E-4,0.123100022598
60722,0.12348787089425196,-0.12358977055149964,0.12773
56740605677,0.12050713415702378,0.12519547592627084,-0
.11895071927886203,0.11725674967757459,0.3085005601304
578,1.5451783392028042,0.0,0.0,0.3222530730629602,1.51
0794060360022,0.019825547669352874,0.01990685598981533
5]},"label":12.35}
```

Modeling results

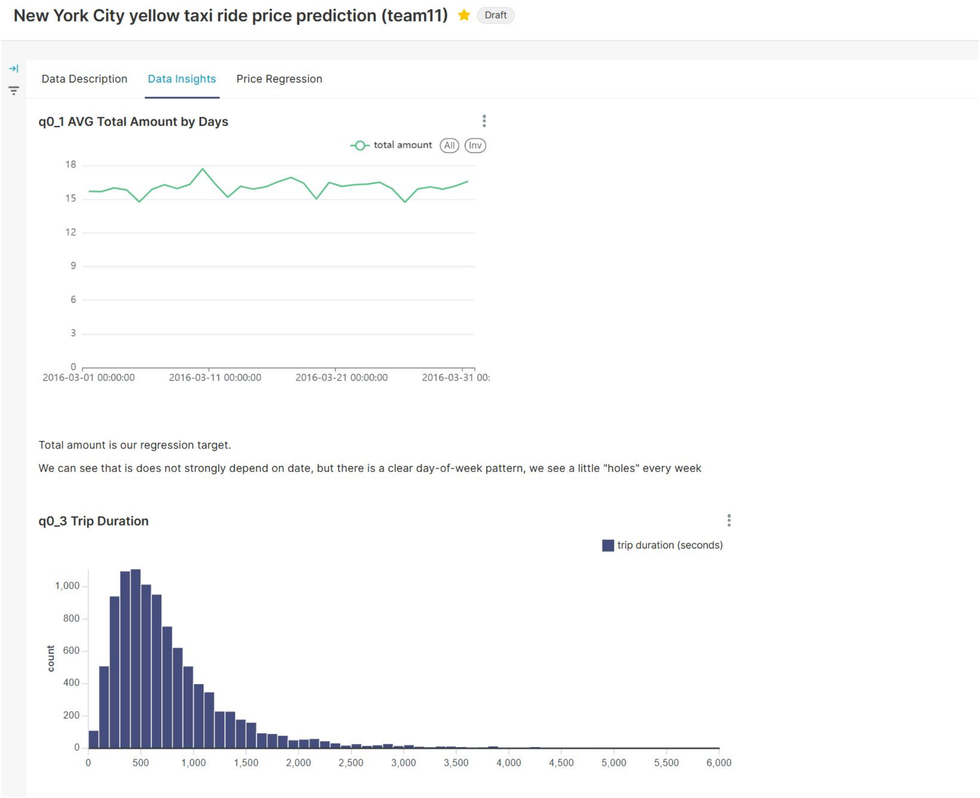
Training on a split train=80%, test=20%
9 768 762 train, 2 442 190 test
Best parameters are <u>highlighted</u>

Model	Parameters space	RMSE base	RMSE tuned	R2 base	R2 tuned	Inference time per sample
Gradient Boosting (GBTRegressor)	maxBins: 24, <u>32</u> maxDepth: 3, <u>8</u>	$5.35	$5.16	0.84	0.85	0.04ms
Random Forest (RandomForestRegressor)	numTrees: 10, <u>40</u> maxDepth: 5, <u>10</u>	$5.77	$5.16	0.81	0.85	0.05ms

Project stages - 4/4 | Presentation

Work done:
Web dashboard in superset describing:

1. Feature types
2. EDA storytelling
3. Model metrics
4. Prediction results



Challenges

1. Cluster uptime
2. Long time of training (~ 4 hours for one grid search run)
3. No git extensions in jupyter and one session for multiple users -> may lead to merge conflicts

Demo

Do you have any questions?