

Fare Enough:

Building ML Models for Predicting Uber Fare Amount

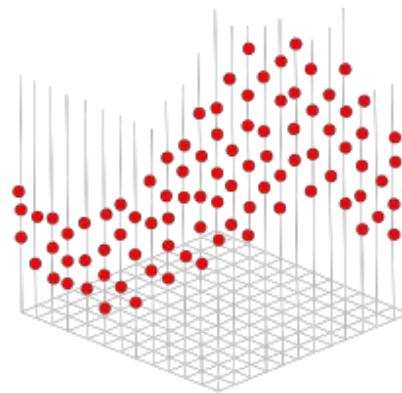
Muchen Zhong

Brown University, Data Science Institute

Oct 25/2024

Github link: https://github.com/muchenzhong/data1030_midterm.git

(including the dataset been used)





Introduction

Intention: Predicting Uber Ride Fare Amount

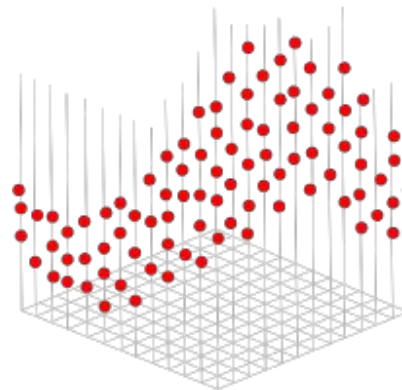
Why This Matters:

- Passengers: anticipating the cost → choose optimal times for rides → cost savings
- Company: strategize its pricing models to maximize revenue during high-demand periods
- Drivers: earning forecasting & optimize trip selection

Type of Problem: Regression

Data Collection:

- Source: Kaggle Platform
- Collected through web scraping from the Uber API.





Dataset Overview

- Large dataset: **190k+** rows, covering the period from 2009 to 2015.
- 6 features + 1 target variable
- No missing values

Pickup _datetime	Pickup _longitude	Pickup _latitude	Dropoff _longitude	Dropoff _latitude	Passenger _count	Fare _amount
2015-05-07 19:52:06 UTC	-73.999817	40.738354	-73.999512	40.723217	1	7.5
2009-07-17 20:04:56 UTC	-73.994355	40.728225	-73.994710	40.750325	1	7.7
...

6 Features

Target Variable



Necessary Data Preprocessing Before EDA



Pick Up DateTime

- Pickup_year
- Pickup_month
- Pickup_weekday
- Pickup_day
- Pickup_hour

A new column added: distance_km

- Calculate the distance between two geographic points (pickup and dropoff locations) using the Haversine formula.



Data Frame Now: 11 features

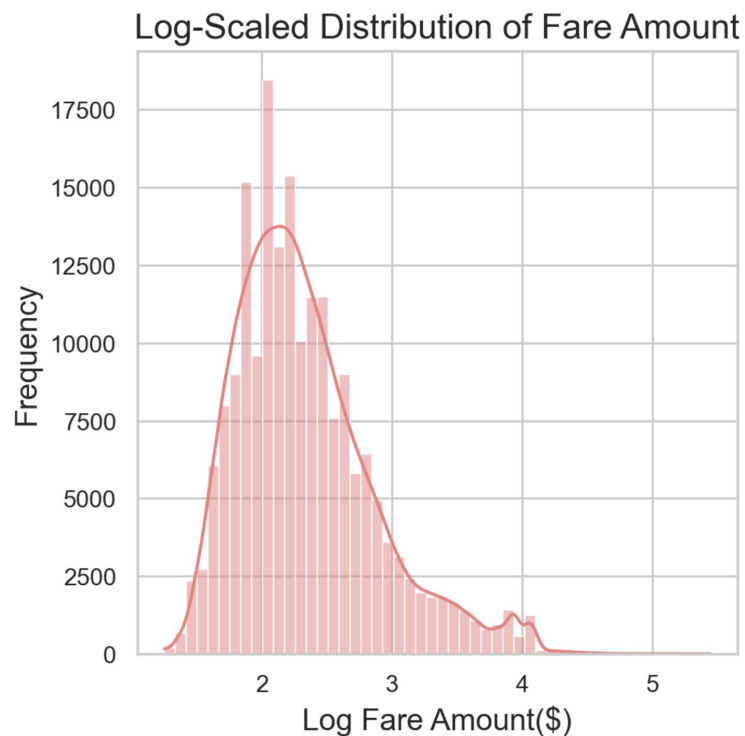
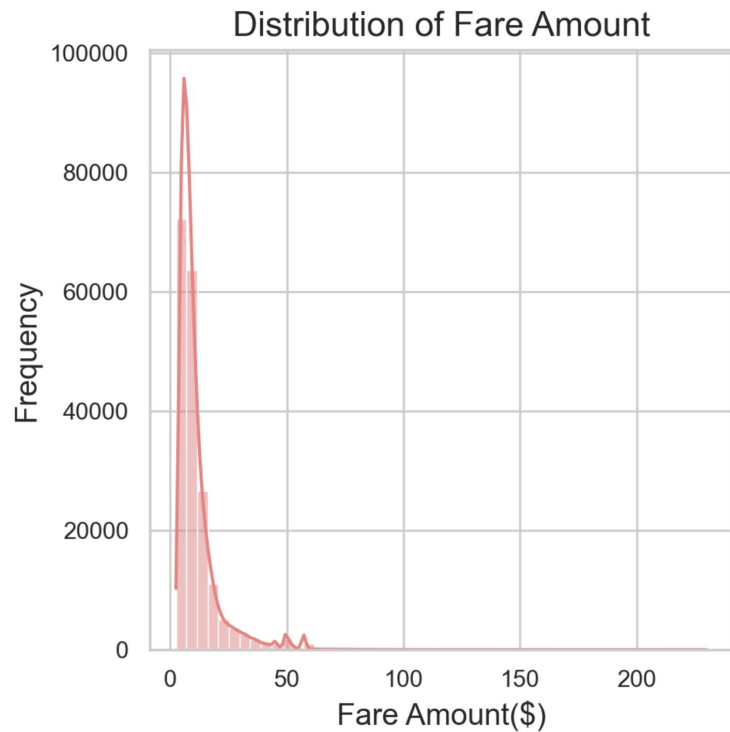
Pickup _datetime	Pickup _longitude	Pickup _latitude	Dropoff _longitude	Dropoff _latitude	Passenger _count	Fare _amount
2015-05-07 19:52:06 UTC	-73.999817	40.738354	-73.999512	40.723217	1	7.5
2009-07-17 20:04:56 UTC	-73.994355	40.728225	-73.994710	40.750325	1	7.7
...



Pickup _year	Pickup _month	Pickup _weekday	Pickup _day	Pickup _hour	Pickup _longitude	Pickup _latitude	Dropoff _longitude	Dropoff_ latitude	Distance _km	Passenger _count	Fare _amount
2015	May	Thursday	7	19	-73.999817	40.738354	-73.999512	40.723217	1.683323	1	7.5
2019	July	Friday	17	20	-73.994355	40.728225	-73.994710	40.750325	2.457590	1	7.7
				



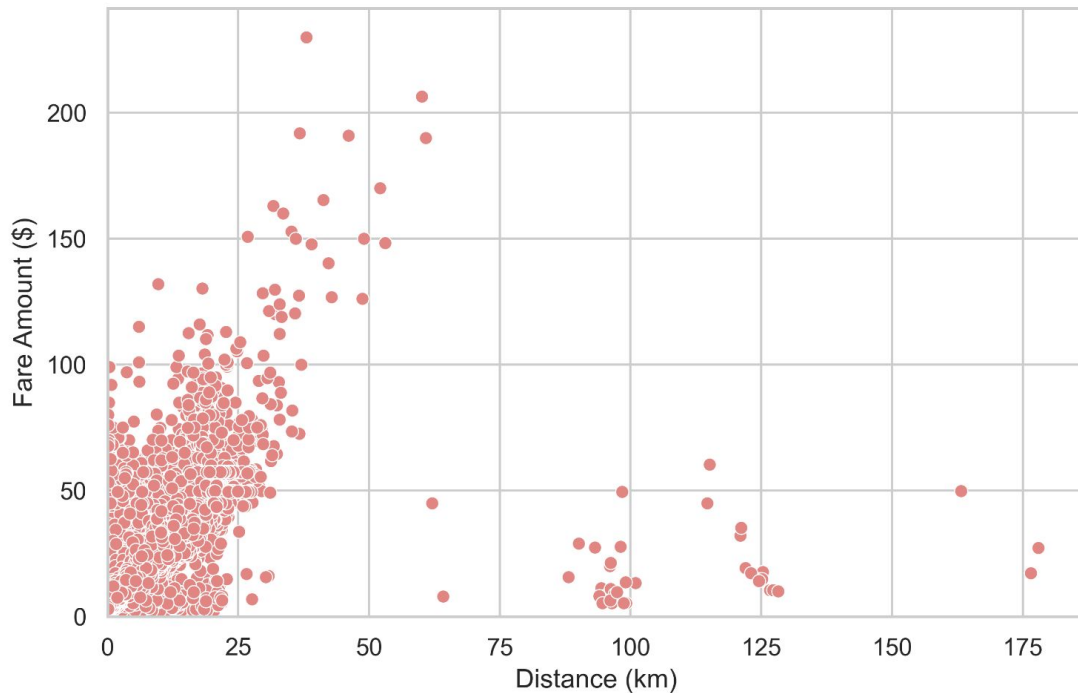
Exploratory Data Analysis





Exploratory Data Analysis

Scatter Plot of Distance vs. Fare Amount



Longer Distance

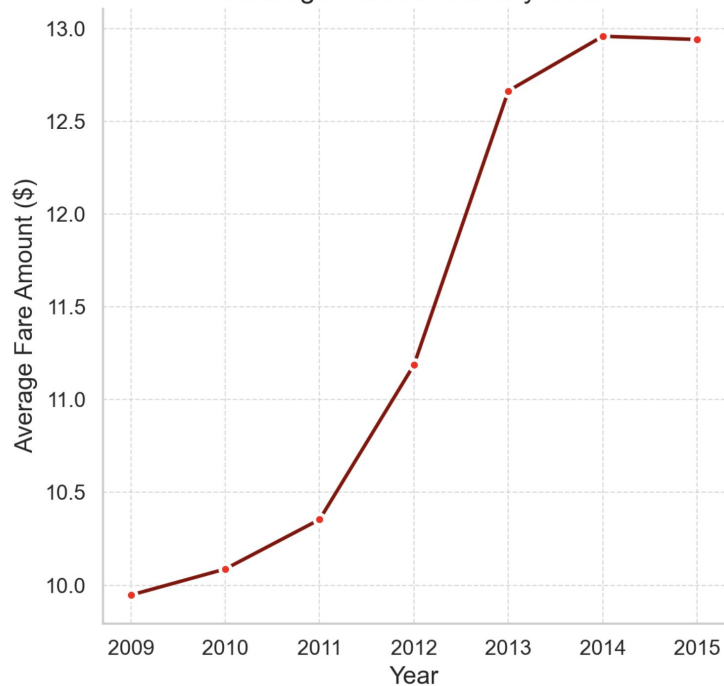


Higher Fare
Amount

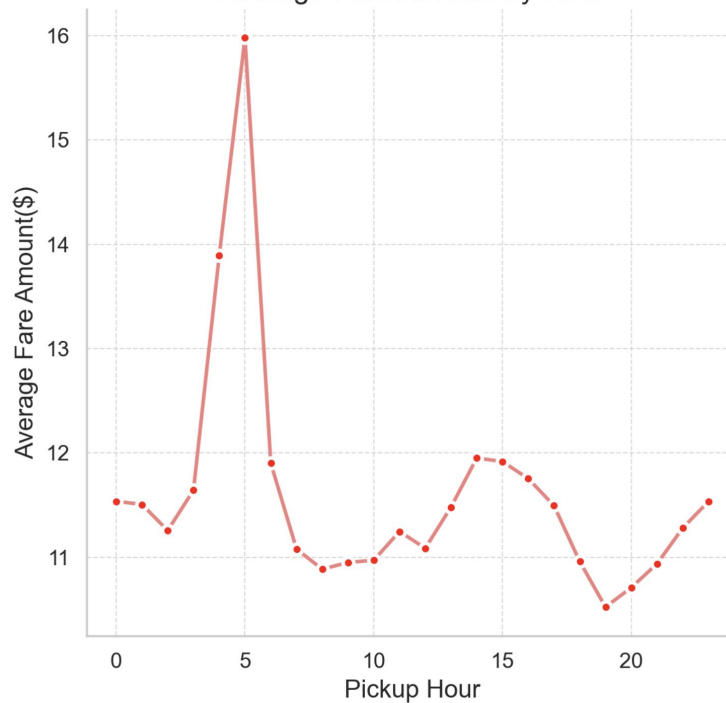


Exploratory Data Analysis

Average Fare Amount by Year



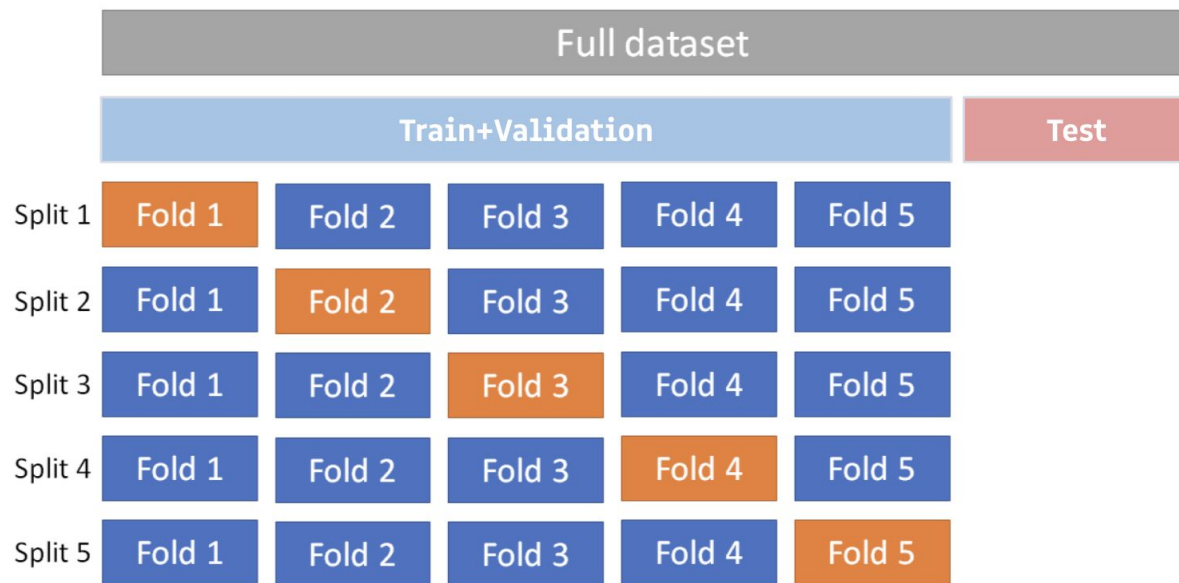
Average Fare Amount by Hour





Data Splitting

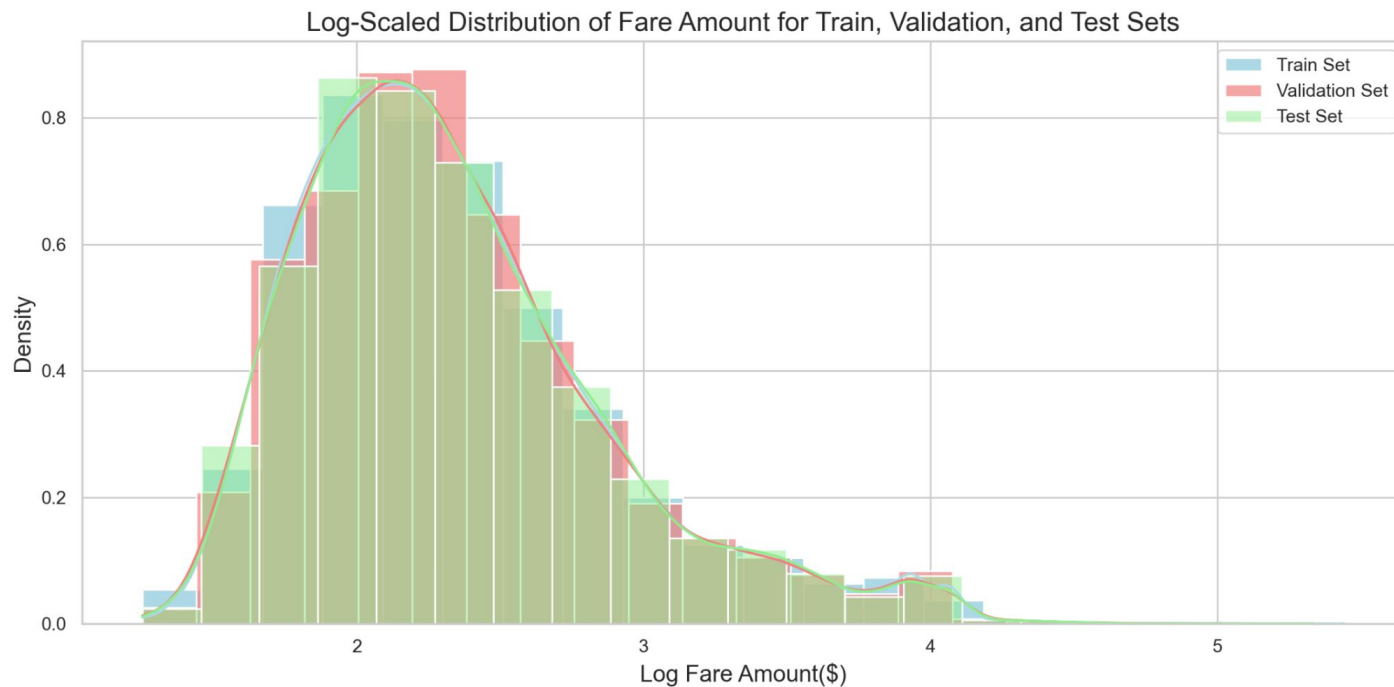
- Shuffle and split out 20% for test set
- Perform a kfold cross validation to split train set and validation set





Data Splitting

- Train, validation, and test set have similar distribution for fare_amount

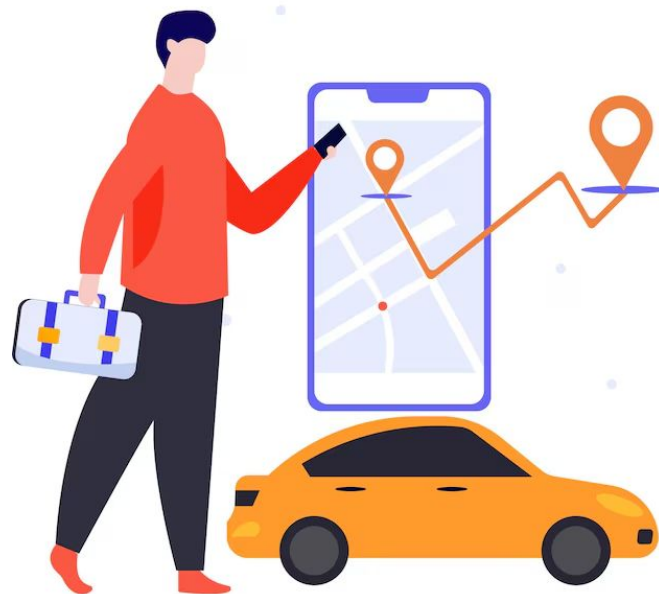




Data Preprocessing: Categorical Feature Encoding

One-Hot Encoding:

- Pickup_month
- Pickup_weekday
- Pickup_day
- Pickup_year





Data Preprocessing: Continuous Features Normalization

MinMax Scalar:

- Pickup_hour
- Pickup_longitude
- Pickup_latitude
- Dropoff_longitude
- Dropoff_latitude

Standardized Scalar:

- Distance_km
- Passenger_count

**11 features → 60 features
ready for the ML Model :)**



References

Ranjan, S. (2020, May 4). K-fold cross-validation in Keras. Medium. Retrieved October 24, 2024, from <https://medium.com/the-owl/k-fold-cross-validation-in-keras-3ec4a3a00538>

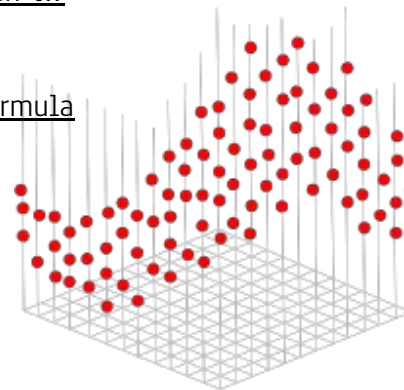
Source: YasserH. (2020). Uber Fares Dataset. Kaggle. <https://www.kaggle.com/datasets/yasserh/uber-fares-dataset>.

Stack Overflow. (2011, March 1). Haversine formula in Python: Bearing and distance between two GPS points. Retrieved from <https://stackoverflow.com/questions/4913349/haversine-formula-in-python-bearing-and-distance-between-two-gps-points>

Wikipedia. (2024, October 21). Haversine formula. Wikipedia. https://en.wikipedia.org/wiki/Haversine_formula

Freepik. (n.d.). Uber png. Retrieved October 24, 2024, from <https://www.freepik.com/free-photos-vectors/uber-png>

Freepik. (n.d.). Uber driver vector. Freepik. Retrieved October 24, 2024, from <https://www.freepik.com/vectors/uber-driver>





Data Science Initiative
BROWN



Thank you

