Practical Approach to Rideshare demand prediction

HoJoon Kim, Master of Science in Information, Daiwei Zhang, Master of Science in Information

Keywords : Machine Learning, Chicago Transportation, Rideshare application, Time Series Analysis

Problem Statement and Objectives

Utilize Data mining techniques to provide insight to rideshare application drivers of Chicago

1. Mine time-series data
2. Build a regression model for ride demands
3. Develop a web application to share the insights

Data Source

Chicago Transportation Network from Oct 2018 to Aug 2020	
Rows	180 Million
Features	21 Features
Data Size	46 GB

Results

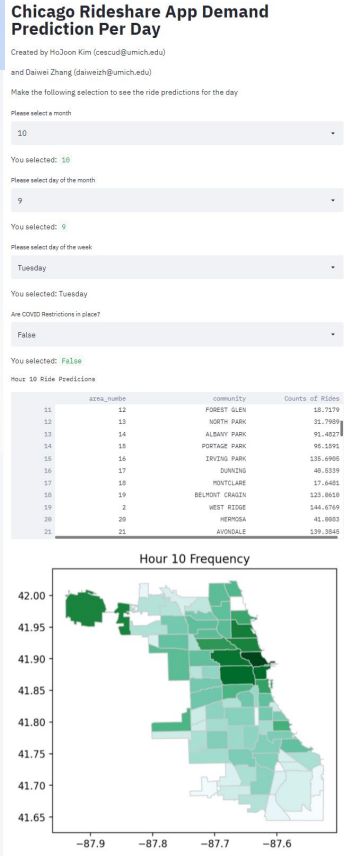
Normal Model

Model	MSE	MAE	R^2
SGD	0.305	0.427	0.170
Decision Tree	0.033	0.133	0.909
Random Forest	0.023	0.110	0.937
Gradient Boost	0.019	0.096	0.948

COVID-19 Restriction Model

Model	MSE	MAE	R^2
Decision Tree	0.025	0.124	0.877
Random Forest	0.024	0.123	0.881
Gradient Boost	0.022	0.115	0.893

Web Application



Methodology

Data Processing

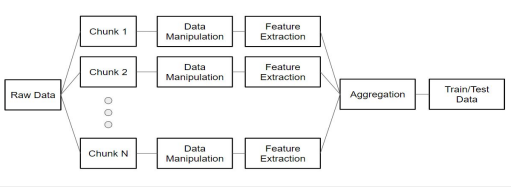
- Process Big Data in partitions using a programmed pipeline
- Convert Time series to Supervised ML Problem
- Aggregate processed data to save cost

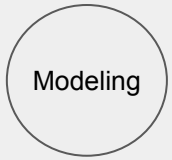
Modeling

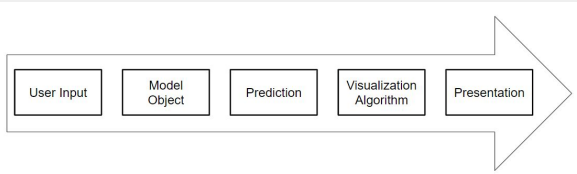
- Preliminary training
- Hyperparameter tuning on promising models
- Model Selection

Data Processing

- Serialize final model for deployment
- User Interface development
- Deploy model for interactive prediction application







Practical Approach to Rideshare Demand Prediction Model of Chicago

University of Michigan
School of Information

Authors

HoJoon Kim, Master of Science in Information
Daiwei Zhang, Master of Science in Information

Problem Statement & Motivation

Due to the global pandemic, there is a significant amount of disruption within the labor market throughout various communities. For the City of Chicago, the unemployment rate has significantly increased from 4.5% in March 2020, to 17.3% in April 2020. Although the unemployment rate gradually decreased, it still remains at 10.8% in Sept 2020 [1]. Chicago had a high number of uber drivers[2] before the pandemic and we expect this number to increase during this period of High unemployment as people look for temporary cash jobs to make their ends meet. We want to provide a free analysis using data mining techniques on Chicago Rideshare app usage data to provide insights to drivers in order to help them maximize their earning potential. Our aim is to build a regression model that predicts the ride demand across different neighborhoods at different times of the year. We would like to develop an interactive web application that the drivers can use to gain quick insights into the demand forecasts.

Methodology

Data Overview

Basic specifications:

Chicago Public Data - Transportation Network Providers - Trips Dataset[3]	
Total number of Rows	180 Million
Total number of Features	21 features
Size of the data	46 GB

Our data was distributed from the City of Chicago public database which included all trip details from all major rideshare applications from November 2018 to August 2020. The features consist of four major types which are time series, geo-spatial, physical, and financial attributes. Times series attributes describe the start and end time of the trip while geospatial attributes provide the coordinates of pick up and drop off locations. Physical and financial attributes cover details such as distances traveled, total fares and amount of tip awarded per trip.

Exploratory Data Analysis

For exploratory Data Analysis, we wanted to look at time-series visualization of the frequency of trips to get a general sense of the patterns in our data and few additional visualizations to see which locations have the highest paying customers.

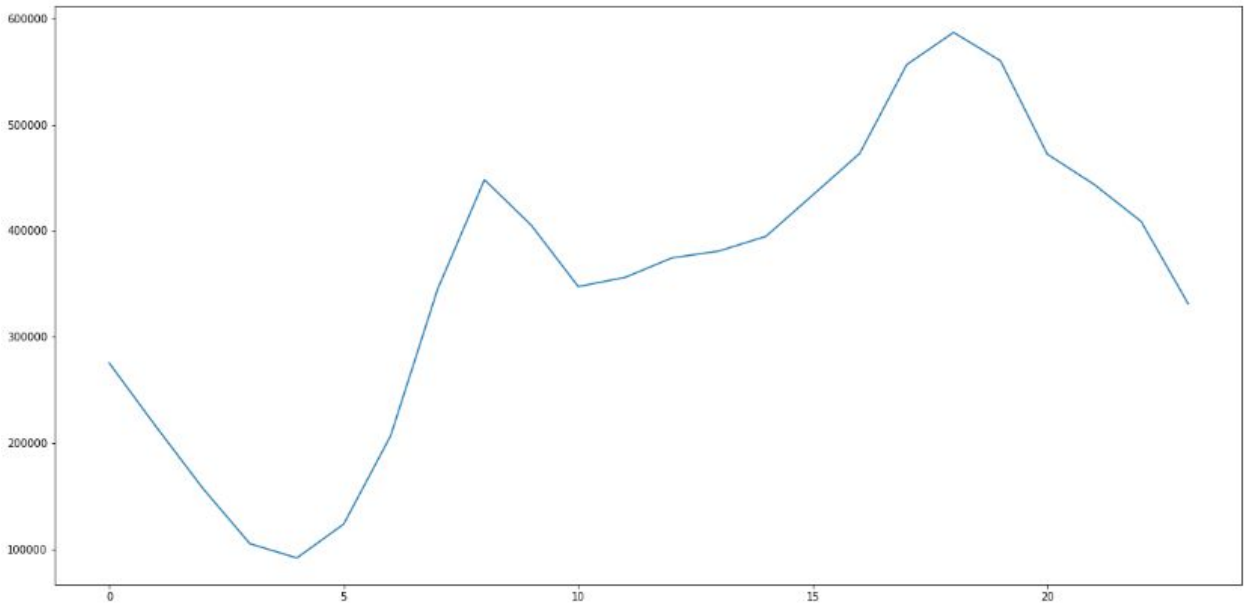


Fig. 1 Ride request per hour in a day (on average)

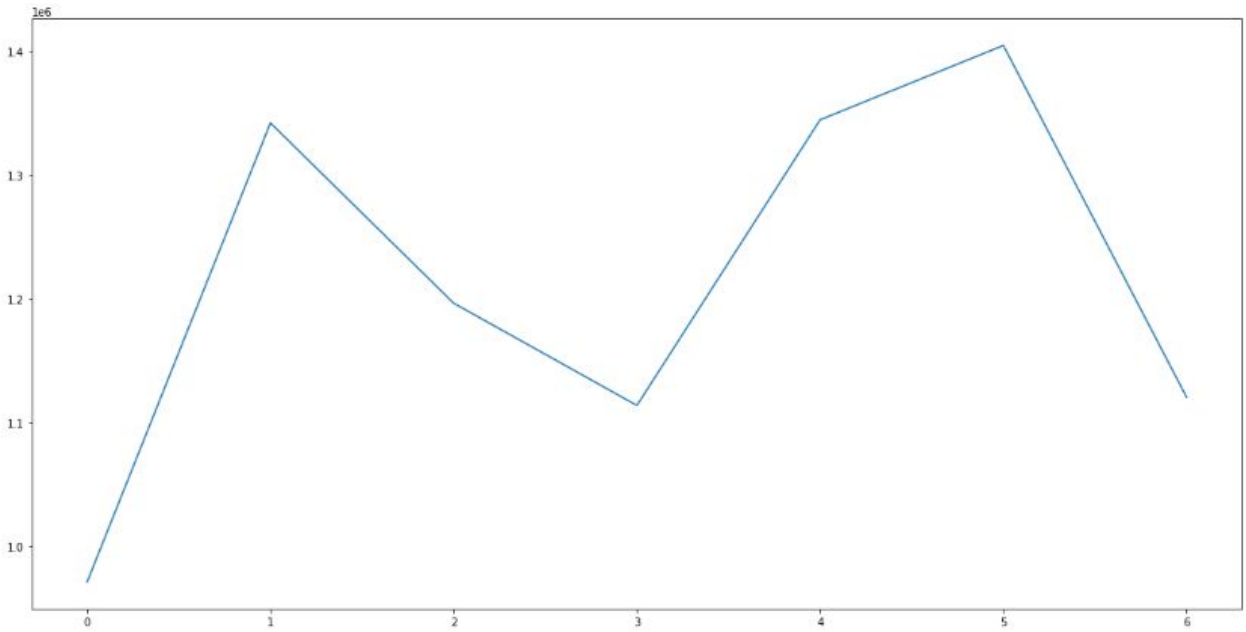


Fig. 2 Ride request per day in a week (on average)

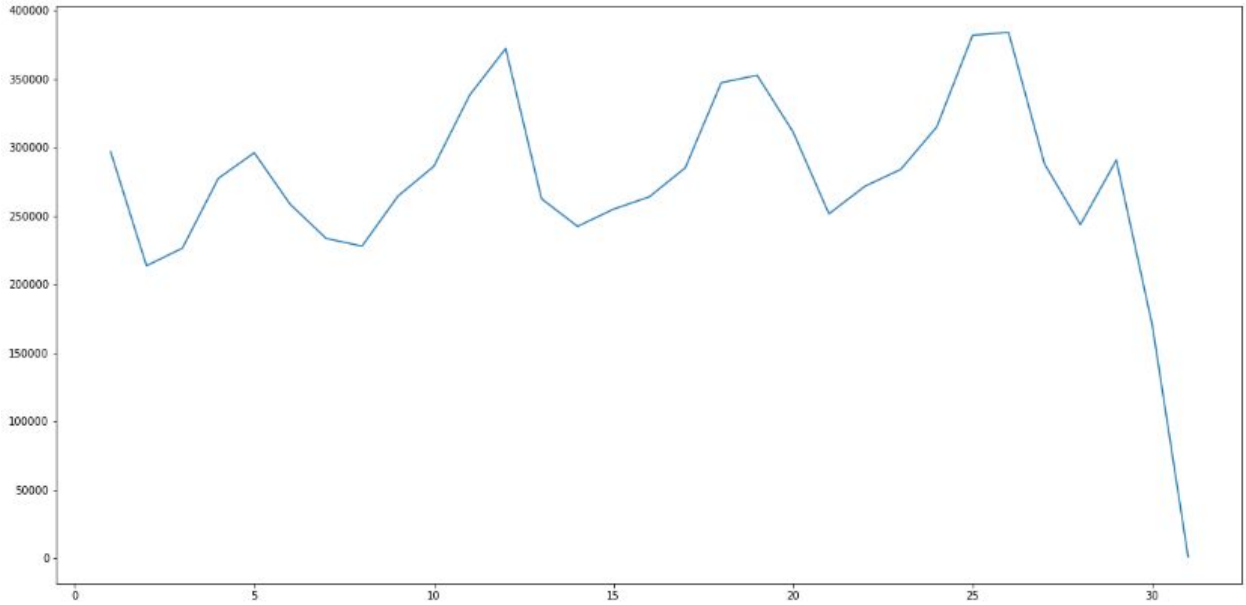


Fig. 3 Ride request per day in a month (on average)

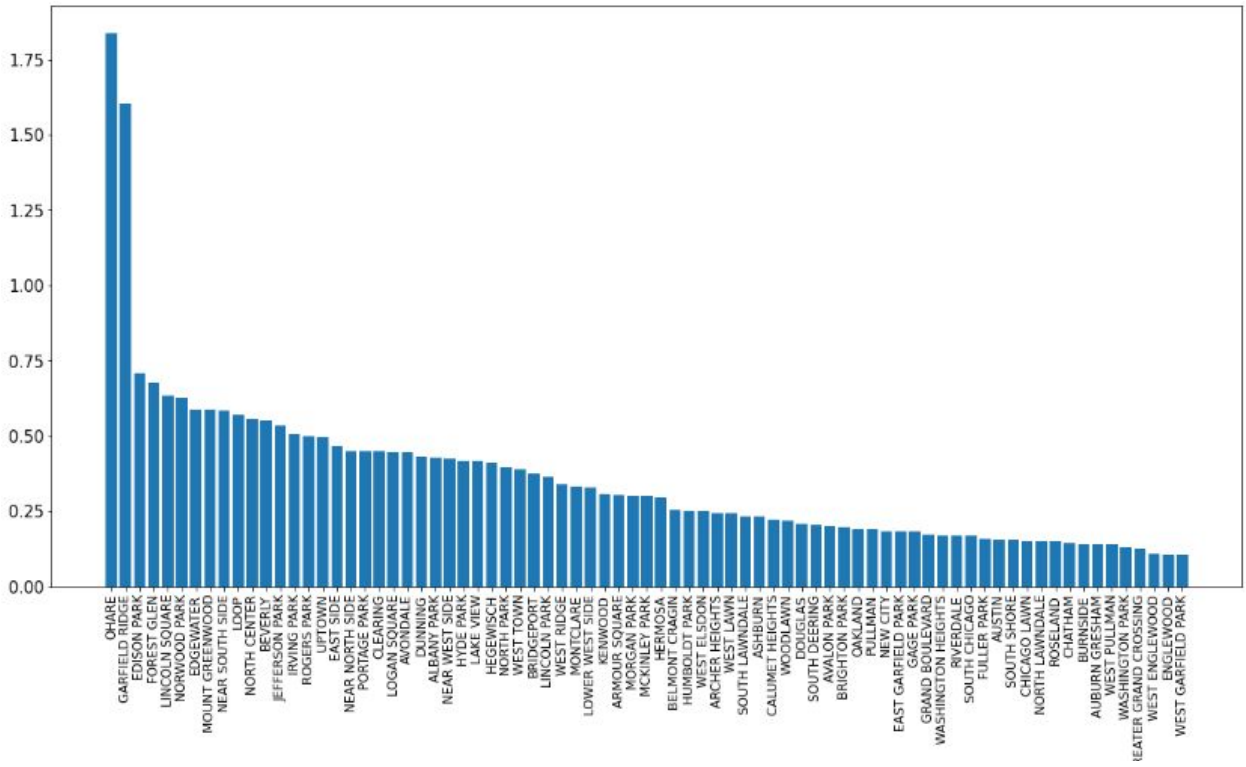


Fig. 4 Highest Tipping Neighborhoods (on average)

Our exploratory analysis results were in agreement with our expectations. The rush hours during the day generated most trips while the later part of the weekdays near weekends generated more trips. There was a cyclical pattern in the month-long visualization which is similar to the pattern we see in the week-long visualization. The neighborhood O'hare had the highest tipping customers and we speculated this was the case because of the Chicago International Airport where customers generally have more items of luggage compared to average customers in other parts of the city.

Data Preparation

Handling of a large dataset

Though not a true big data, our rideshare dataset was fairly large to work on personal computing devices. The file size exceeded 40GB which made it impossible to load the data in its entirety due to memory limitations. We had to create a data processing pipeline that processes the data in chunks while carrying out data manipulation per chunk for the output of the pipeline to be ready for modeling.

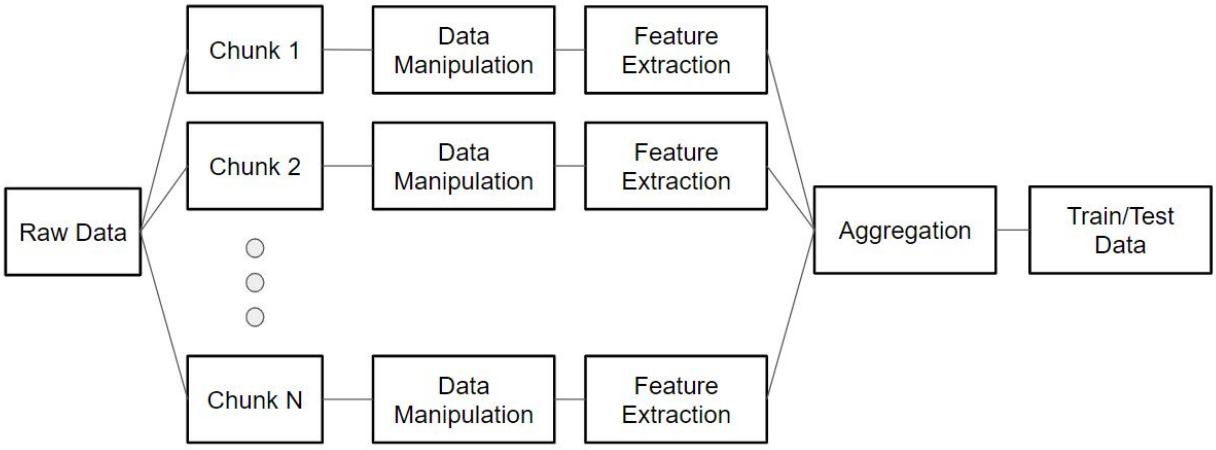


Fig. 5 Data Processing Pipeline

Feature Extraction

The main goal of the feature extraction was to convert the time series data to a supervised machine learning problem while adding additional features that will allow us to create a model

that produces predictions for ride demand throughout different regions of the city. Since we only have one complete year within the time range of our dataset, it was not applicable to use the year as a feature. We used the ride request timestamp to extract month, day of the month, day of the week, and hour of the day as input variables for our training and test data. The minutes were discarded for computing efficiency and the records were grouped together by the months, dates, days of the week, and hours. By doing so we were able to create our output variable total counts of rides per hour. We also incorporated neighborhood codes of Chicago (total of 77) to keep the location information intact in our input variables. In the end, we had 77 output values per hour of any given day and the month.

Further processing

Our exploratory data analysis revealed, at least for the normal times without COVID-19 restriction, the distribution of counts of rides per hour showed extreme right-skewed distribution. This was because of ride requests being congested mostly around the main downtown area especially during rush hours. We converted the counts of rides to a log scale to increase the performance of our regression models. Lastly, all input features have been normalized using the standard scaler from the sci-learn library.

Train and Test data

We wanted to have an entire year as our training data which will allow us to create predictions for any month and date in our application even with models like random forest regressors which suffer when introduced to the problem of extrapolation[4]. Therefore, we used the entirety of 2019 data as our training data.

Modeling

We decided to create two different final models. This was inevitable because of the complete rideshare app demand pattern change that happened due to the COVID-19 outbreak. If we look at Illinois's response to COVID-19, the lockdown restriction took place on March 15th and ended on May 29th[5]. Using a model trained by demands during the pre-COVID-19 times seriously undermined the performance of the model when predicting demands when COVID-19 restrictions were in place.

Therefore our normal model will use the entirety of 2019 data to train and use November and December of 2018 and January and February of 2020 to test the performance. Our COVID-19 restriction model will use March and April of 2020 data to train and use May of 2020 to test the performance.

Multiple regression models will be trained at their default parameters to carry out the preliminary modeling on the data. Some models will serve the role as baseline models for comparison

purposes, while other ensemble models will be further tuned depending on the initial performance. The models that we will explore are linear regression, stochastic gradient descent regression, random forest regression, AdaBoost regression and gradient boosting regression.

After the preliminary training for both situations, random forest regression and gradient boosting regression showed the most promising results. The hyperparameters of the two models have been tuned while using cross-validation to prevent overfitting. For random forest models, minimum sample split, minimum samples leaf, max depth and the number of estimators have been tuned. For gradient boosting model, loss, minimum samples leaf, minimum samples split, max depth, and the number of estimators have been tuned.

Results and Discussion

Normal Model Regression Report

Regression Model	Mean Squared Error	Mean Absolute Error	r_squared score
Ordinary Least Squares	0.303	0.425	0.175
Decision Tree	0.033	0.133	0.909
Stochastics Gradient Descent	0.305	0.427	0.170
Adaboost	0.165	0.328	0.551
Random Forest	0.023	0.110	0.937
Gradient Boosting Regressor	0.019	0.096	0.948

COVID-19 Restriction Model Regression Report

Regression Model	Mean Squared Error	Mean Absolute Error	r_squared score
Decision Tree	0.025	0.124	0.877
Adaboost	0.124	0.289	0.388
Random Forest	0.024	0.123	0.881
Gradient Boosting Regressor	0.022	0.115	0.893

As shown in the regression report above, for both models, the gradient boosting regressor performed the best with a mean squared error value of 0.019 for the normal model and 0.022 for the COVID-19 restriction model. We used mean squared error as our metric to determine the

best model. We can see that the linear regression-based model does not perform very well as you can see with the poor performance of ordinary least squares and the stochastic gradient descent model. This is mainly because of the locational feature (neighborhoods) in our input data that has no linear relationship with our target variable (counts of rides). Amongst the tree-based methods, the more robust models which are random forest and gradient boosting regressor performed better than the baseline model of decision tree regressor. Overall, our final model of choice was gradient boosting regressor with the following parameters: loss=huber, max depth = 10, minimum samples leaf = 2, minimum samples split = 2, and the number of estimators = 100.

Web Application Development

Data mining is different from simple machine learning in that it seeks to discover patterns and models that are valid, useful, unexpected, and understandable[6]. Our modeling and analysis were done to accomplish the first three qualities. We would like to apply the insights gained in the previous section to provide actionable insights that are easy to understand. We will achieve this goal by developing an interactive web application that our main viewers, drivers of rideshare applications, can use to make data-driven decisions to maximize their earning potential.

Model export for deployment in the web application

We used the package called Pickle in order to serialize and save our best model as an object that can be imported and used to create predictions within a python application. Both the scalers and the models were exported as 'normal_model.sav', 'normal_scaler.sav', 'covid_model.sav', and 'covid_scaler.sav'.

Python application development platform: Streamlit

We used the Streamlit package that enables us to effectively design a web-based python application especially for creating visualizations. We created a user interface where the user is able to input a month, a date, day of the week, and the presence of COVID-19 restrictions to get the demand prediction of different neighborhoods of Chicago by the hour of the day. The predictions will be presented in a form of a geospatial heatmap and raw table that shows the counts of rides as the demand.

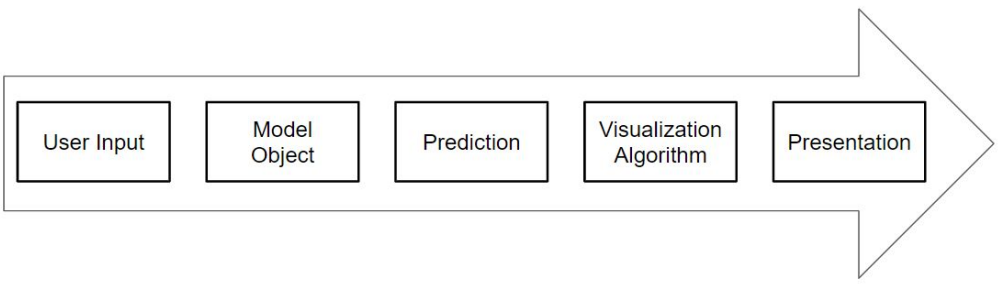


Fig. 6 Web Application Process

Fig. 7 Web Application Interface (Normal Model)

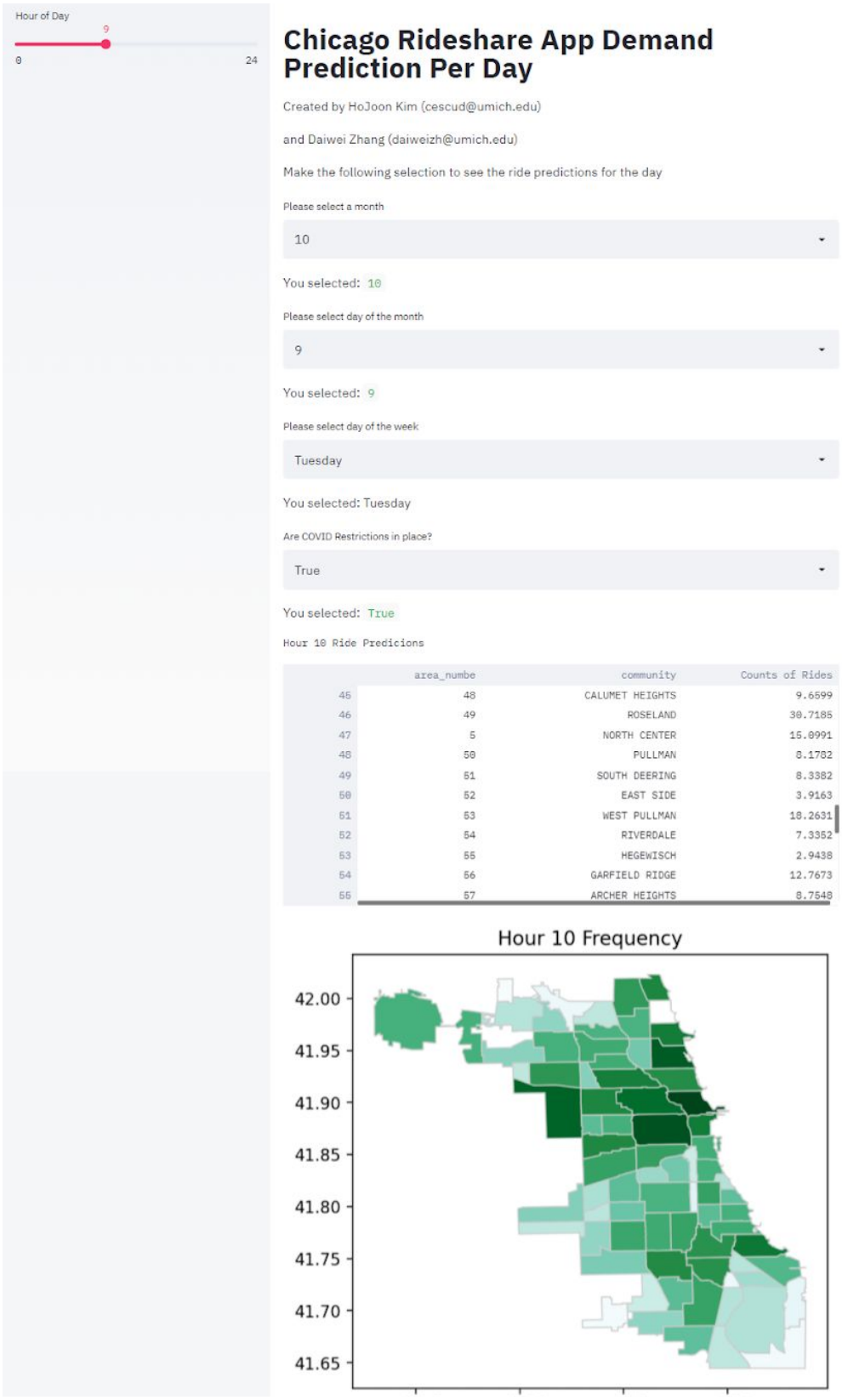


Fig 8. Web Application Interface (COVID-19 Model)

How to run the Application

1. Packages required:
    - Streamlit, Pandas, Numpy, Geopandas, Pickle, Matplotlib
2. Unzip the web_application.zip
3. Run the below command in the command prompt while setting the working directory to the unzipped folder:
    - streamlit run ride_application_chicago.py

Conclusion and Future Work

The motivation for this project came from high-quality rideshare application trip data from the City of Chicago public database. We wanted to discover how the demand of the trips varied across time, and across different neighborhoods in the city. We also wanted to develop an easily understandable platform to share the insights that we discovered with our intended audience; the drivers of rideshare applications in the City of Chicago. We hope that our contribution will bring value to the hard-working drivers in this unprecedented and challenging time.

We learned that when geographic features are included in the input data of our model, tree-based methods performed better due to the lack of linear relationships between geospatial attributes and the demand. With hyperparameter tuning and cross-validation, our final model was the gradient boosting regression model which showed the highest performance in terms of a mean squared error value.

For future work, there are numerous ways that our model and application can be improved to increase the value they provide to our intended audience. For example, we can incorporate more features such as weather variations and explore other features that can contribute to the performance of our model. In addition, we can increase the detail of our predictions by using more pin-pointed geo-spatial units rather than entire neighborhoods which will allow the drivers to narrow their desired location down to even specific blocks within the neighborhoods. This will obviously require more powerful computing architecture. We should explore hiring cloud instances in services like Amazon Web Services to make this high level of analysis possible.

Acknowledgments

We would like to express our gratitude to Professor Paramveer Dhillon and the teaching team of the SI671 Data Ming course for guiding us through this project.

Work Cited List

[1] “Chicago : Midwest Information Office : U.S. Bureau of Labor Statistics.” *U.S. Bureau of Labor Statistics*, 10 Sept. 2014, https://www.bls.gov/regions/midwest/il_chicago_msa.htm.

[2] Channick, Robert. “Too Many Uber Drivers? Chicago Cabbies and Ride-Share Workers Join Forces, Urge Cap on Uber and Lyft Cars - Chicago Tribune.” *Chicagotribune.Com*, Chicago Tribune, 30 Oct. 2018, https://www.chicagotribune.com/business/ct-biz-chicago-taxi-ride-share-drivers-limit-20181030-story.html.

[3] “Transportation Network Providers - Trips | City of Chicago | Data Portal.” *City of Chicago | Data Portal | City of Chicago | Data Portal*, 2 Oct. 2018, https://data.cityofchicago.org/Transportation/Transportation-Network-Providers-Trips/m6dm-c72p.

[4] Hengl, Tomislav & Nussbaum, Madlene & Wright, Marvin & Heuvelink, Gerard & Graeler, Benedikt. (2018). Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables. PeerJ. 6. e5518. 10.7717/peerj.5518.

[5] “City of Chicago :: COVID-19 Orders.” *City of Chicago*, https://www.chicago.gov/city/en/sites/covid-19/home/health-orders.html. Accessed 8 Dec. 2020.

[6] Dhillon, Paramveer. “Data Mining: Methods and Applications.” SI671 Data Mining. University of Michigan, 31 Aug. 2020, University of Michigan, Ann Arbor. Lecture.