Oren Shapira

CS 677 Python for Data Science

**Machine Learning Regression Models for Predicting Uber and Lyft Prices:**

**Dataset Description**

This dataset, pulled from [Kaggle](https://www.kaggle.com/ravi72munde/uber-lyft-cab-prices)) has ~700,000 records of Uber and Lyft rides with information on price, surge-multiplier for price, time, distance, weather, and other various data features. As many have experienced first-hand, Ride Share prices tend to surge when there is high rider demand and/or low supply of drivers. While distance of the ride is surely a major predictor of the Ride Share fare, my hypothesis is that weather (i.e temperature, wind, etc.), time of day, and day of the week will also have an impact on rider demand - and therefore prices.

**Project Overview and Objective:**

The primary purpose of this project was to build several machine learning algorithms for predicting the price of Uber and Lyft ride fares. Testing of various algorithms would help me uncover the relative impact of different predictor variables on price. The models I explored in-depth were simple/multivariate/polynomial regression, k-Nearest Neighbor Regression, Random Forest Regression, and Gradient-Boosted Regression. I also briefly tested Decision Trees and Ada-Boost Regression.

A secondary purpose of this project was to automate the creation of these models against user-defined cross-stratifications of the dataset. For example, I wanted to have the flexibility to see how these models performed on only Uber rides vs. only Lyft Rides, among only ride types that are "XL". Once these (and other) stratum parameters are defined, the notebook would recreate all exploratory analysis, visualizations, and ML model creations - each being optimally hyper-tuned. The final result would be a data frame that compare various performance metrics of each regression model. With minimal modifications, this notebook can hopefully serve as a template for regression analysis done on other datasets as well.

**Key Project Notes:**

My final project deliverable is a Jupyer notebook with code and in-depth commentary on functions and models. After initial data exploration of the raw dataset, I chose to subset my data as described below. The commentary written within the notebook is specific to these parameters, but they can easily be updated at the beginning of the Jupyter notebook. If the notebook is re-executed, all data analysis, modeling, and hypter-tuning would be re-done to fit the dataset subset defined.

1. I filtered only on rides with surge pricing:
   1. One of the dataset variables is surge-pricing multiple (e.g. 1x, 2x, 2.5x, etc.) I had initially assumed that surge values would have a somewhat normal distribution, but the raw data had a very small proportion of surge-priced records. My concern was that this dataset over-represented rides with that didn’t have ‘surged’ ride demand – and that distance may end up being the only relevant predictor of price. In order to gauge whether other factors impacted rider demand, and therefore price, I focused my analysis on only rides with a surge-multiplier > 1.0.
      1. An alternative would have been to set surge-multiplier or price-per-mile as the target variable. But my ultimate goal was to predict price, which I felt would be a more easily interpretable result.
2. Cab-type:
   1. I only filtered on ‘Lyft’ rides, because these were the only ones that had any surge-priced rides in the dataset.
      1. I left this strata as an option in case analysis were to be re-done that wasn’t filtered on ‘surged’ rides.
3. Ride-type:
   1. I only filtered on ‘Standard’ rides, because I felt that mixing in ‘Luxury’ or ‘Pool’ (which have different pricing rates) would adversely affect my model’s predictive power.
      1. I left this strata as an option in case analysis were to be re-done on a different ride-type (i.e. maybe ‘Luxury’ rides are more affected by weather than ‘Standard’ rides).
4. Predictors:
   1. My predictor variables were temperature, precipitation level, humidity, wind speed, time of day, and day of the week.
5. Target variable: Price of ride
6. Sample size of dataset:
   1. I looked at 100% of the dataset, which was already reduced to 20,975 records because of my surge-pricing filter.
      1. I left this strata as an option to help reduce computational processing that would have been needed on the full large dataset.

**Instructions for Running Code:**

* Option 1: The zipped project folder contains the Jupyter notebook and raw datafile.
  + Unzip the datafile, run the Jupyer notebook, and make sure the filepath for reading the .csv file is updated within the notebook
* Option 2: View on [GitHub](https://github.com/omshapira/Uber_Lyft_Analysis/blob/master/Ride%20Sharing%20Prices.ipynb): A couple of the interactive charts may not display
* Option 3: View Git NBViewer Rendered [version](https://nbviewer.jupyter.org/github/omshapira/Uber_Lyft_Analysis/blob/92f3493b26108f4c4310d95164d77c1e4b5744cb/Ride%20Sharing%20Prices.ipynb): One of the interactive widgets for backward elimination won’t be able to display
* Option 4: From the rendered version above, you have the option to download the notebook, or execute code via a Git Binder (will take a few minutes to load). These options are the two icons at the top right of the page.
  + - Filepath for datafile would still need to be updated.
    - Changes to notebook via binder will not affect project posted in Git repository.

**Libraries Used:**

* Numpy
* Pandas
* Os
* Datetime
* Matplotlib
* Seaborn
* Sklearn
  + Submodules include: train\_test\_split, metrics, LinearRegression, standardScaler, LabelEncoder, neighbors, KNeighborsRegressor, RandomizedSearchCV, GridSearchCV, ensemble, DecisionTreeRegressor, GradientBoostingRegressor, and AdaBoostRegressor
* Scipy
* Statsmodels.api
* \_future\_
* Ipywidgets
* IPython
* plotly
* math