PRICE PREDICTION OF USED CARS

ORIE 4741 PROJECT MIDTERM REPORT

Jiahe Chen *
jc3472@cornell.edu

Wei Cheng †
wc655@cornell.edu

November 2, 2021

1 Data pre-processing

The original dataset contains 400,000+ rows and 26 columns. Since the dataset was scraped from Craigslist, some columns does not contain any information: id, url, region_url, image_url, VIN, country. After dropping those columns, we have 19 features and 1 output to predict. Those features contains numerical, categorical and text data. Since we have a very large amount of data points, we also dropped all the rows with NA data. So, we finally got a dataset with 79016 rows and 20 columns.

2 Data Description

Fig. 1a plots the population of missing values for top 10 features that have most missing values. We can see that size has the most missing data.

In terms of the quantities and basic stats of the data, after pre-processing, we finally got a dataset with 79016 rows and 21 columns. The 'price' column is the output we want to predict. We then split our dataset into training set (80%) and test set (20%).

Detailed descriptions of all 19 features are listed below. To help us better understand the data, we also visualize some critical features in Fig. 1.

- **price**: numerical data. Fig. 1b shows the distribution.
- **region**: categorical text data.
- **manufacturer**: categorical text data.
- **model**: categorical text data. Indicates the model of the car, such as highlander, elantra, etc.
- **condition**: categorical text data. Fig. 1c shows that most cars are in 'good' or excellent' condition.
- **cylinders**: categorical text data. Car engine with more cylinders can provide more power but also consume more gas. Most cars have 4 to 8 cylinders as seen in Fig. 1d.
- **fuel**: categorical text data. Most cars run gas as seen in Fig. 1e.
- **drive**: categorical text data. Fig. 1f shows the distribution.
- **lat & long**: numerical data. The latitude and longitude of the location of the car. Fig. 1g shows that most cars are from California and Northeastern regions.
- **odometer**: numerical data. Indicates how many miles traveled by the car. Fig. 1h shows the distribution.
- **title_status**: categorical text data. Most cars have clean title as seen in Fig. 1i.
- **transmission**: categorical text data. Most cars have automatic transmission as seen in Fig. 1j.
- **size**: categorical text data. It contains 4 categories - 'full-size', 'compact', 'mid-size', 'sub-compact'.

*School of Electrical and Computer Engineering, Cornell University
†Ann S. Bowers College of Computing and Information Science, Cornell University

- **type**: categorical text data. Fig. 1k shows the distribution.
- **year**: numerical data. Fig. 1l shows that most cars are manufactured in 2000 to 2020.
- **paint_color**: categorical text data. It contains 12 colors.
- **description**: text data. The text description of the car.
- **state**: categorical text data. The state of the location of the car.
- **posting_date**: text data. It indicates when the car information was posted.

3 Project Plan

In terms of the data encoding, since there are a lot of categorical features in the dataset, we initially planed to use one-hot encoding. However, we later found that the feature vector size can go to 9000+, which will cause over-fitting, and make it extremely slow to train the model. So, now we are looking into ways to use embedding to encode those data, so that we can have a smaller feature vector.

To avoid over-fitting, we will try to use embedding to encode the model feature, and balance the number of data points and features. We can also restrict model complexity by parameter setting.

Since there are abundant features in our data, it's generally less likely to under-fit. However, if it happens, we can add new features or increase the model complexity.

4 Preliminary Analyses

We trained some OLS models by using different combinations of features to see how they impact the training results.

1. Dataset with only real-value features.
   We trained a model with only numerical features - year, odometer, lat, long. We got training MSE $\sim 24 \times 10^{13}$ and test MSE $\sim 18 \times 10^{11}$. Clearly, just include numerical feature is not enough to fit the data.
2. Dataset with categorical features.
   There are a lot of categorical features in our dataset. Some features are very important, such as model and manufacturer. However, if we tried to use one-hot encoding for all categorical features, the feature vector size will be 9028. We tried to run a OLS model, but it was never ended.
3. Dataset with categorical features excluding model feature.
   After a closer look at categorical data, we found that the model feature has 8475 categories. So, although model is supposed to be a very important feature; if we can exclude it, our feature vector size can be reduced to 553. We ran another OLS model and got training MSE $\sim 23 \times 10^{13}$ and testing MSE $\sim 10 \times 10^{11}$. We can see there's an improvement from the previous model.

5 What to do next

1. Find a better way to encode model data.
   As mentioned above, there are currently 8475 models; if we use one-hot encoding, the feature vector will become too large. We can try to use embedding to reduce the size of the feature vector. Another potential way to address this is to use the price of the new car to replace the model feature. However, this will require us to find a new dataset.
2. Use more complex model.
   We currently only tried OLS model. However, without a regularize, OLS model can over-fit easily. We can later try some more complex models, like ridge regression or lasso regression; and figure out which model gives the best performance.
3. Try train separate models for each car model.
   Since each car model has very different price range, instead of just train a single model, we can try to train separate models for each car model.
4. Tune the model parameter by cross validation.
   Currently, we are using the default parameters of the model. Later, we can do cross validation to tune the model parameters, and find out the best model.

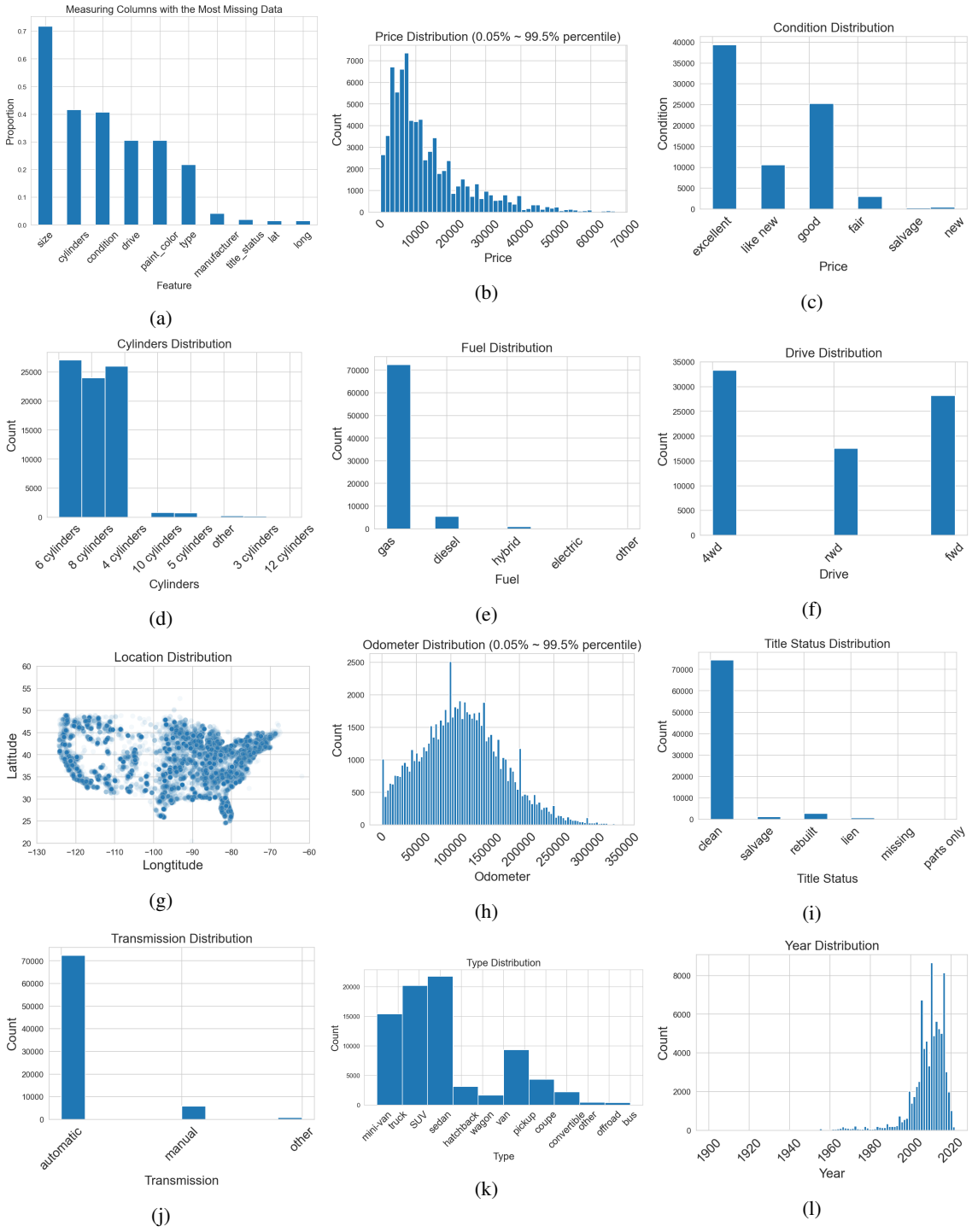

Figure 1: Visualizations of the data.