Used Car Price Prediction

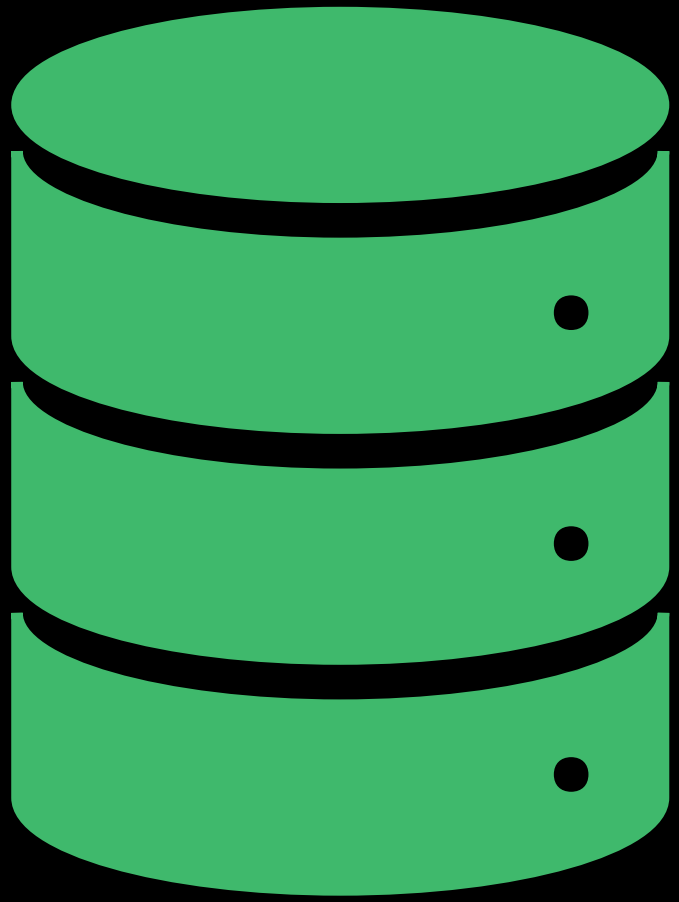
Group 7
Zehao Liu 193074000
Jialong Zhang 190227130

Goal

- To find the best performing model from multiple training models using a processed dataset
- To be able to predict the price of used cars by a set of attributes

Dataset

- We use the train-data.csv from https://www.kaggle.com/datasets/avikasliwal/used-cars-price-prediction, which contains 14 attributes.



```
Data columns (total 14 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Unnamed: 0         6019 non-null   int64
 1   Name               6019 non-null   object
 2   Location           6019 non-null   object
 3   Year               6019 non-null   int64
 4   Kilometers_Driven  6019 non-null   int64
 5   Fuel_Type          6019 non-null   object
 6   Transmission       6019 non-null   object
 7   Owner_Type         6019 non-null   object
 8   Mileage            6017 non-null   object
 9   Engine             5983 non-null   object
 10  Power              5983 non-null   object
 11  Seats              5977 non-null   float64
 12  New_Price          824 non-null    object
 13  Price              6019 non-null   float64
```

Data attributes

- Get by train_data.info()

Processing the data

- Handle missing data and redundant attributes
- Dealing with outliers
- Handle categorized data

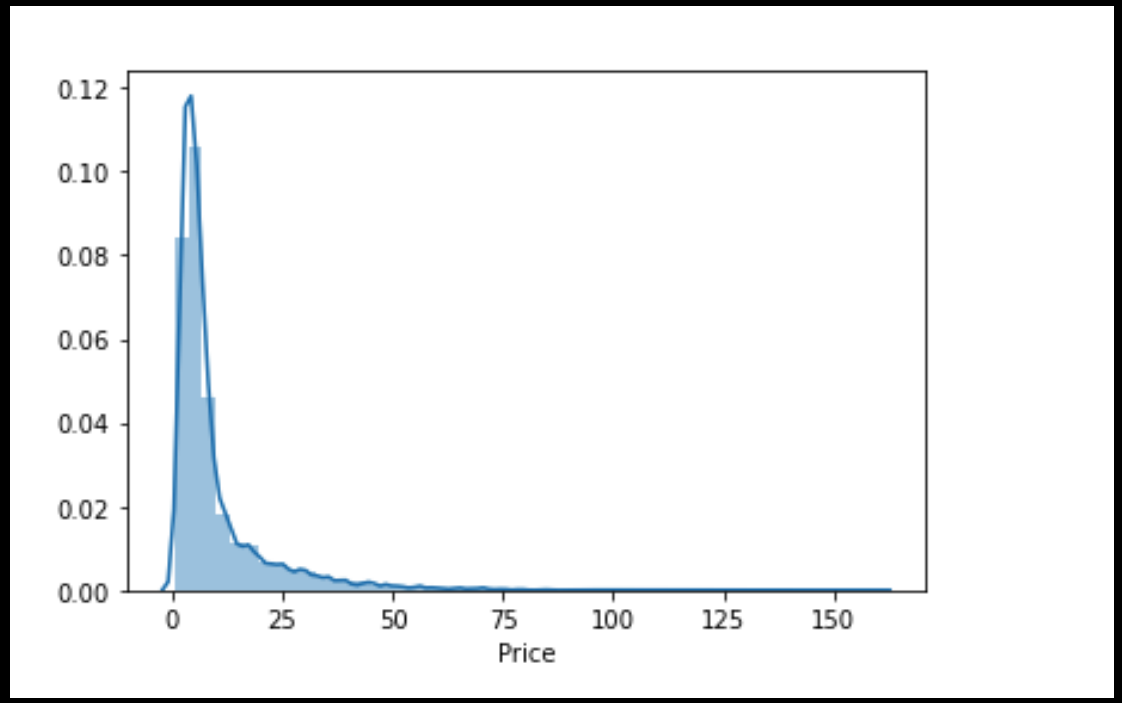
Handle missing data and redundant attributes

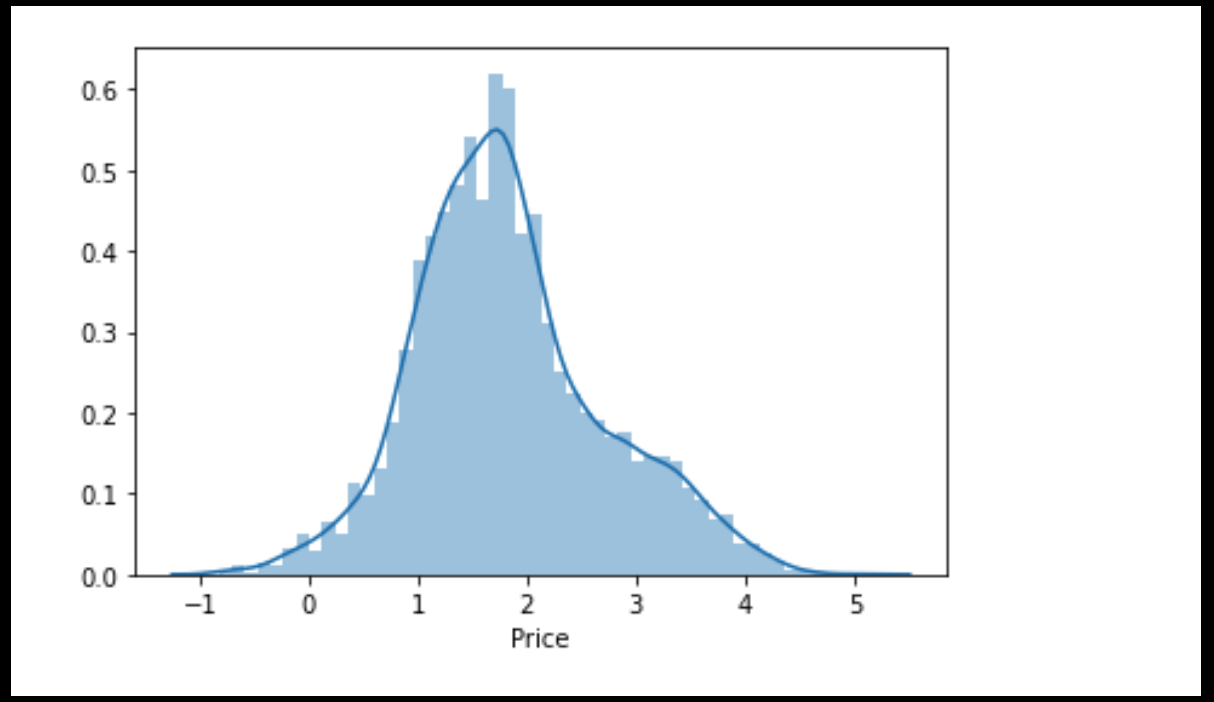
```
data.isnull().sum()
```

```
Unnamed: 0            0
Name                  0
Location              0
Year                  0
Kilometers_Driven     0
Fuel_Type             0
Transmission          0
Owner_Type            0
Mileage               2
Engine               36
Power                36
Seats                42
New_Price          5195
Price                 0
dtype: int64
```

```
print(train_data.isnull().sum())
train_data.drop(["New_Price"],axis=1,inplace=True)
train_data = train_data.dropna(how='any')
train_data = train_data.reset_index(drop=True)
```

Dealing with outliers

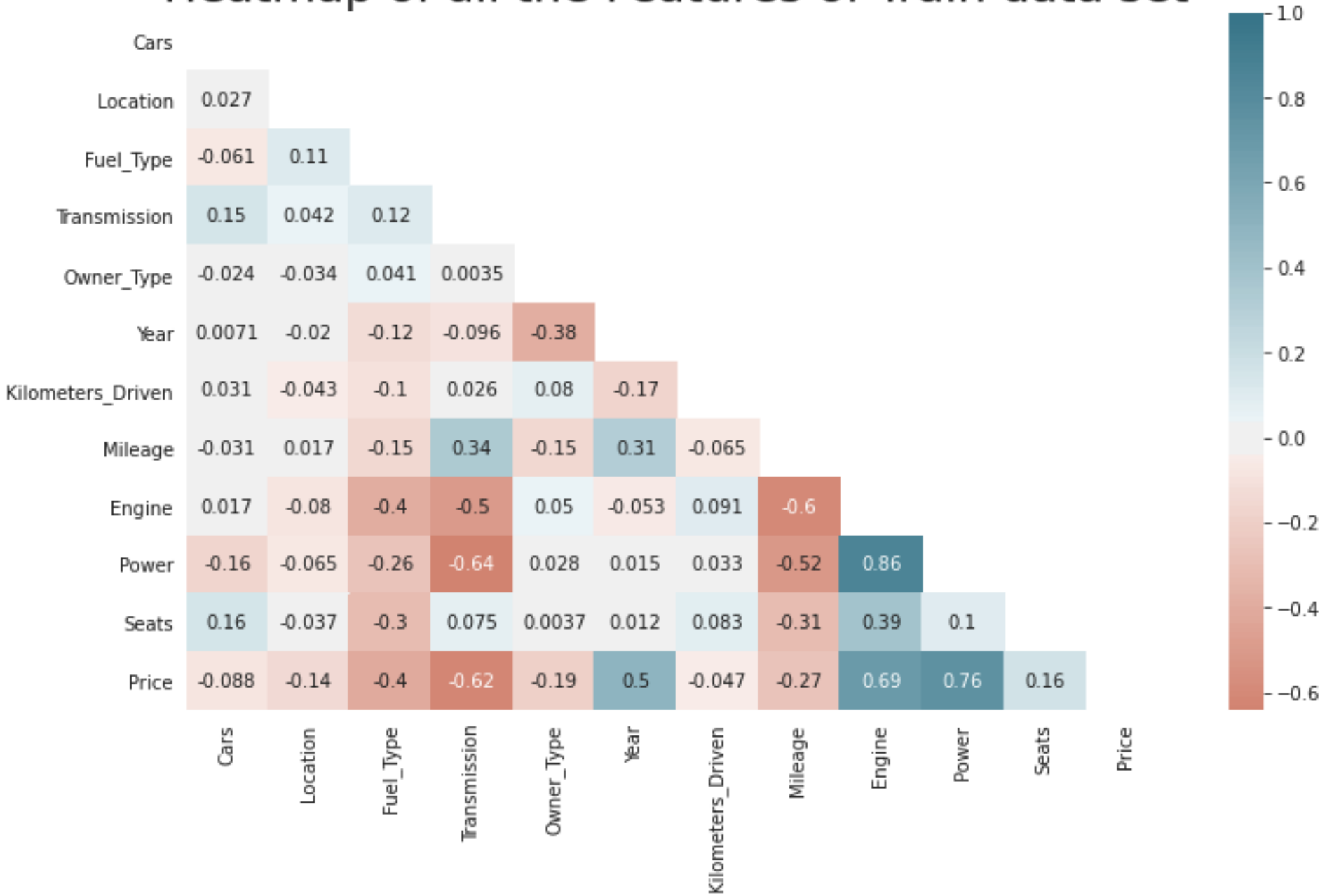




Handle categorized data

```
#Handling Categorical parameters
from sklearn.preprocessing import LabelEncoder
label_encoder = LabelEncoder().fit(data['Cars'])
data['Cars'] = label_encoder.transform(data['Cars'])

label_encoder = LabelEncoder().fit(data['Location'])
data['Location'] = label_encoder.transform(data['Location'])

label_encoder = LabelEncoder().fit(data['Fuel_Type'])
data['Fuel_Type'] = label_encoder.transform(data['Fuel_Type'])

label_encoder = LabelEncoder().fit(data['Transmission'])
data['Transmission'] = label_encoder.transform(data['Transmission'])

label_encoder = LabelEncoder().fit(data['Owner_Type'])
data['Owner_Type'] = label_encoder.transform(data['Owner_Type'])
```

Heatmap of all the Features of Train data set



Build Models

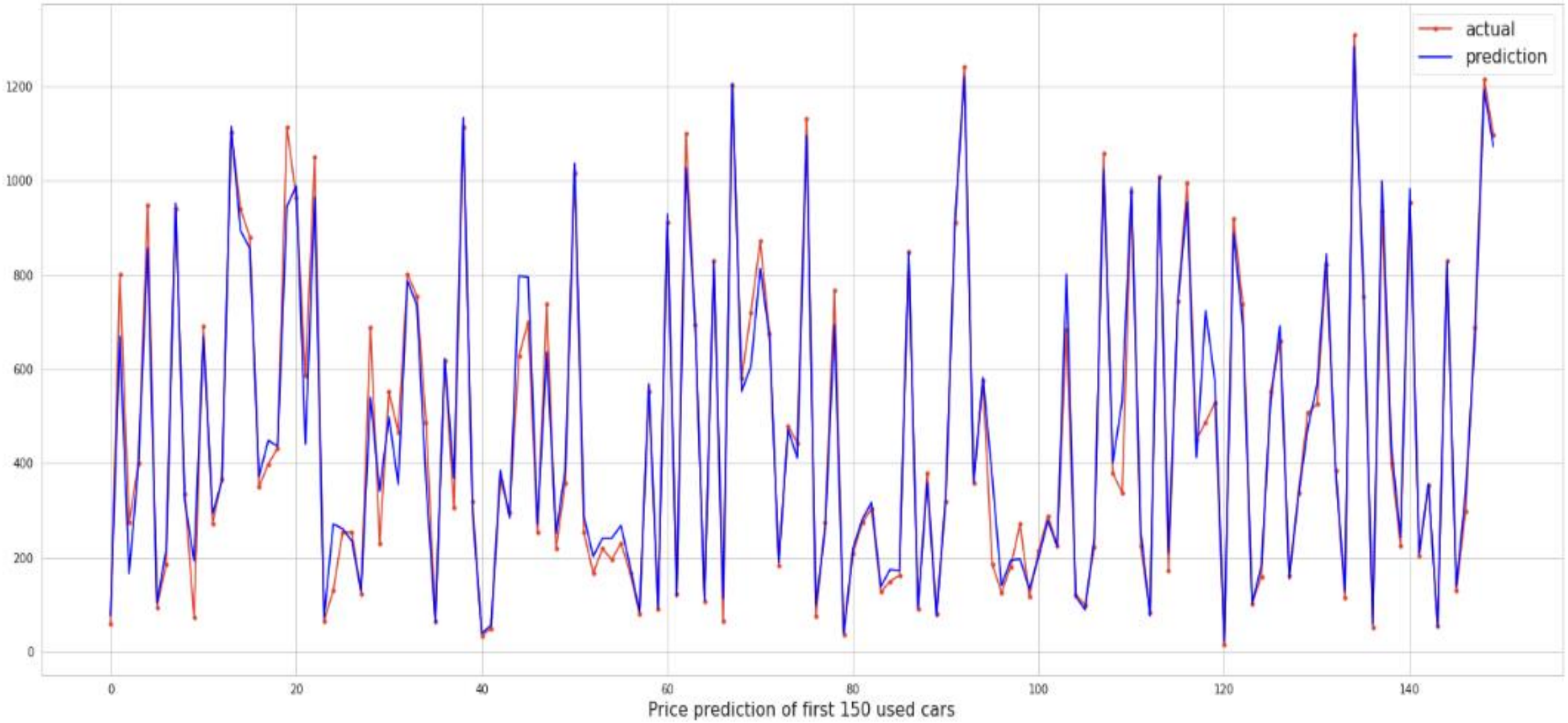
```
#Build Model
models = [['DecisionTreeRegressor', DecisionTreeRegressor()],
          ['RandomForestRegressor', RandomForestRegressor()],
          ['LinearRegression', LinearRegression()],
          ['KNeighborsClassifier', KNeighborsClassifier()],
          ['DecisionTreeClassifier', DecisionTreeClassifier()],
          ['Multilayer perceptron', MLPClassifier()]]
```

Accuracy Ranking

	model	Root Mean Squared Error	Accuracy on Traing set	Accuracy on Testing set
5	Multilayer perceptron	288.758081	0.013990	0.004564
3	KNeighborsClassifier	354.978130	0.210092	0.005578
4	DecisionTreeClassifier	136.206343	0.997502	0.025862
2	LinearRegression	152.635554	0.847900	0.805983
0	DecisionTreeRegressor	112.634383	0.999993	0.894350
1	RandomForestRegressor	84.589637	0.991733	0.940411

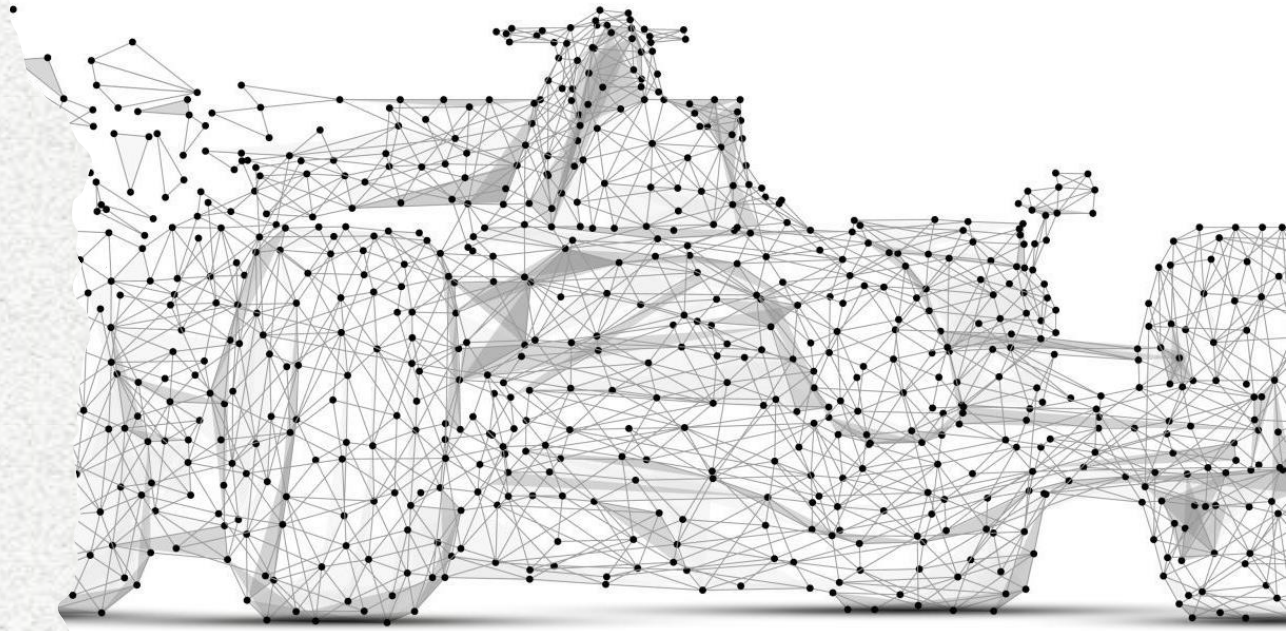
Error Table

```
                Error Table
Mean Absolute Error          :  56.038213713450816
Mean Squared  Error          :  6242.634232221394
Root Mean Squared  Error     :  79.01034256489079
Accuracy on Traing set       :  0.992622121565873
Accuracy on Testing set      :  0.9480127802728119
```


actual
prediction
1200
1000
800
600
400
200
0
0
20
40
60
80
100
120
140
Price prediction of first 150 used cars

FUTURE WORKS

Based on the current model, we intend to develop an interactive system that will enable users to input information to determine the appropriate price for a used car.



Contribution

- Jialong Zhang:
- Processing the data
- Data Visualization

- Zehao Liu:
- Build Models

Thank you for watching