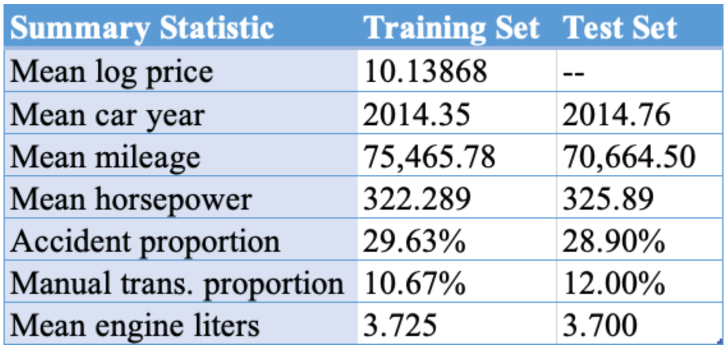
Three Rivers Auto Technical Report

Final Data Science Project by Danny Kennedy

Introduction/Exploratory Data Analysis

This analysis attempts to predict the list price of a used car as accurately as possible based on 15 different predictors. The predictors in this dataset are a mix of numerical variables including a car's horsepower, mileage, and number of engine cylinders, as well as categorical variables, such as the brand of the car and its exterior color. The available training data consists of 1,809 used cars, along with their price, which has been log-transformed using a natural logarithm. The provided testing set has data on 1,000 used cars, but without a known price.



Summary Statistic	Training Set	Test Set
Mean log price	10.13868	--
Mean car year	2014.35	2014.76
Mean mileage	75,465.78	70,664.50
Mean horsepower	322.289	325.89
Accident proportion	29.63%	28.90%
Manual trans. proportion	10.67%	12.00%
Mean engine liters	3.725	3.700

The table above shows some important summary statistics for the dataset. For the most part, the data in the test set have very similar attributes to the data in the training set, meaning it is relatively safe to assume that a model built on the training data will perform similarly well on the testing data. One notable difference between the two data sets is the fact that the mean mileage differs by roughly 5,000 miles. Since a car's mileage is expected to be inversely related to its price, the average price of a car in the test set is likely slightly lower than in the training set. Furthermore, the actual average price in the training set is $25,303 when un-transformed, which seems to be a fair average price for a used car in the United States.

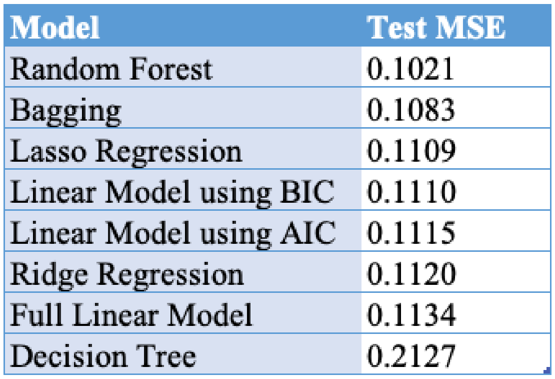
In order to combat the number of categorical variables present in the dataset, I created dummy variables for the *accident*, *fuel_type*, and *transmission_type* variables. I kept the existing categorical variables for the car's brand and color because of the large number of categories present in each column, but had to address them when fitting the lasso and ridge regression models later on. One oddity present in the training data is that the car with id=275 is an extreme outlier. This car, which is a 2005 Maserati with 32,000 miles, has a selling price of $2,954,086, which is considerably larger than the second highest selling price in the training data of $324,996. It is unclear why this car's selling price is so high, as there are several other similar Maseratis in the dataset that have a selling price nowhere near it. This, combined with a very extreme Cook's distance value, led me to remove it from the training set. Lastly, there was no missing data in either file, meaning that the analysis could be carried out using all of the remaining observations.

Methods Overview/Details

In total, I tested eight different models on the data. My general idea was to start with the most basic models and work my way down to the most complex. I therefore started by fitting some simple linear models, but then moved on to shrinkage techniques as well as tree-based methods. The first model that I fit was a simple linear regression with all of the available predictors. This initial model gave a good idea of which predictors were the most significant as well as how much signal there was in the data. I then moved on to using forward selection with two different selection criteria in order to build a more refined linear model. For the first one, I used Akaike information criterion (AIC), and for the second I used Bayesian information criterion (BIC), meaning I would end up with two linear models of different sizes: the second would have fewer predictors since the BIC penalty is higher for larger data sets. I then moved onto regularization methods in order to see if adding some bias would help my predictions. For this, I fit a lasso model and a ridge regression model on the data while tuning the λ shrinkage parameter for both models using cross validation. Finally, I moved on to using tree-based methods for prediction. I started with just one simple regression tree, but then used bagging to build a forest of one-hundred regression trees in order to make the predictions more robust and less overfit. For my last model, I fit a random forest model to see if some bias in the form of randomness would ultimately lead to a more accurate result. For this model, I tuned the mtry parameter using the out-of-bag (OOB) error to find the optimal forest.

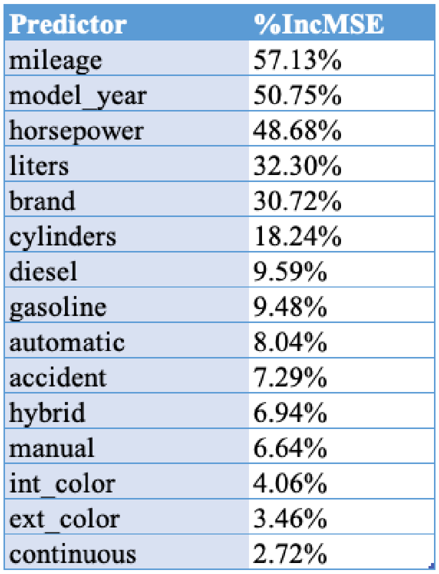
Summary of Results

Firstly, it is important to note that there is a high signal-to-noise ratio present in this dataset. The full linear model that was fit at the very beginning has a multiple R-squared of 0.8584, which is very high for real-world data. This means that most models should do quite well at making predictions, which was reflected in the results. In order to test the predictive accuracy of each model, I split the available training data into a “train_train” training set and a “train_test” validation set, and found the mean squared error (MSE) of each model using these two sets. The resulting test MSEs are summarized below.



Model	Test MSE
Random Forest	0.1021
Bagging	0.1083
Lasso Regression	0.1109
Linear Model using BIC	0.1110
Linear Model using AIC	0.1115
Ridge Regression	0.1120
Full Linear Model	0.1134
Decision Tree	0.2127

The most accurate model was ultimately determined to be the random forest model, which performed the best when its mtry parameter was set to four. Importantly, however, most of the models performed similarly, with a less than ten percent difference in the predictive accuracy of the full linear model compared to the optimal random forest model. This shows how little noise is present in the data, since overfitting did not seem to be a major issue at all for any of the models, with the exception of the basic decision tree. Furthermore, when tuning the λ shrinkage parameter for the lasso model, it was found that the optimal λ was 0.00346, which is a small number corresponding to very little shrinkage at all. This would only happen in a data environment with lots of signal and not much noise. The most surprising result was the fact that the random forest model outperformed bagging. Random forests have more bias than bagged trees do (since mtry < p), which is typically a bad thing when the data quality is high. Ultimately, there are enough significant predictors in this dataset to make this increase in bias worthwhile.



Predictor	%IncMSE
mileage	57.13%
model_year	50.75%
horsepower	48.68%
liters	32.30%
brand	30.72%
cylinders	18.24%
diesel	9.59%
gasoline	9.48%
automatic	8.04%
accident	7.29%
hybrid	6.94%
manual	6.64%
int_color	4.06%
ext_color	3.46%
continuous	2.72%

The above table shows the importance of each individual variable, organized from the most important to the least important for the optimal random forest model. I used %IncMSE to measure variable importance, which determines how much the MSE would increase if a given predictor’s values were randomly permuted, since I used MSE as my measure of accuracy throughout the whole analysis. Overall, the most important variables are unsurprising: everyone would expect that a car’s mileage and year would be highly correlated with its price. However, a few variables did stand out, particularly the fact that a car’s brand is only the fifth most important factor, as well as the fact that a car’s exterior color is the second *least* important factor in determining selling price. Furthermore, all of the models agreed that mileage, model year, and horsepower were the three most important features driving price. Past the top three, however, different models value different predictors more, with the tree-based models valuing liters and cylinders much more than the linear models, for example.

Conclusion/Takeaways

In conclusion, it is safe to assume that the three most important factors in determining a car’s selling price are its mileage, year, and horsepower. Furthermore, it can be concluded that a car’s exterior and interior color do not actually have a significant impact on its selling price, despite their seeming importance. The final random forest model can be trusted because the residuals have no discernible pattern and follow a roughly normal shape, plus the predicted price of each car differs by only about 10.7% from the actual price. While this result is not perfect, there is not much more that a model can do, as there is always some irreducible error in a dataset that cannot be explained. In the future, it would be useful and interesting to see how these variables would play out on a dataset of just brand new cars, since neither model year nor mileage—the two most important variables in this analysis—could be used. In that case, I would expect the brand of a car to become much more important in determining selling price.