# Summary of findings

## Models Used

The “Module 11 Workbook.ipynb” contains 10 different models. They are described below.

First, five models use ‘price’ as the target variable. The resulting training ‘mean\_squared\_error’ (MSE) in each of the models is very high (of the order of 10^14).

The next five models use natural logarithm of ‘price’ denoted as ‘log\_price’ which resulted in reducing the MSE to a single digit.

Given below are short descriptions of each of the last five models that use log\_price. The same description is provided in “Module 11 Workbook.ipynb”.

### Model #6

It uses ncyl (‘cylinders’ converted to integer) and log\_price and LinearRegression model

Traning MSE taken on the whole dataframe is mod6\_mse = 6.8816110644533195

### Model #7

It uses ['condition', 'transmission', 'drive', 'type', 'ncyl'] with condition as ordinalencoder

transmission, drive, type as onehotencoder, ncyl in PolynomialFeatures object (degree = 3)

as part of a pipeline

Training MSE taken on the whole dataframe is mod7\_mse = 1.3119076332608168

### Model #8

It uses 1 ordinalencoder, 8 onehotencoders and 4 PolynomialFeature objects (ncyl, odometer, year, f\_state (derived from frequency of each state)) with ncyl having degree = 3 and remaining degree = 1

Training MSE taken on the whole dataframe is mod8\_mse = 1.668658711858038

Model #8 has total 94 features as a result of various encoders and PolynomialFeatures

After doing the analysis of p-values (< 0.05) and t-statistics, both yielded the same set of 7 most significant features given below

Most significant features

col# col\_name

90 polynomialfeatures-1\_\_ncyl^3

91 polynomialfeatures-2\_\_year

92 polynomialfeatures-3\_\_odometer

93 polynomialfeatures-4\_\_f\_state

0 ordinalencoder\_\_condition

89 polynomialfeatures-1\_\_ncyl^2

88 polynomialfeatures-1\_\_ncyl

#### Please see the workbook for the code

### Model #9

It uses the Sequential feature Selection on model #8 (mod8) features

The input dataframee is shuffled and split into 80% train and 20% test indices

n\_features\_to\_select = 7

MSE is 1.3144517741100517

### Model #10

Model 10 uses GridSearchCV and Ridge regressor along with the same ColumnTransformer as in model 8 (94 features)

Best value of alpha is 1e-05

Corresponding Test MSE is 1.191933129669491

Corresponding training MSE is ~ 1.225

The plots of Test MSE and training MSE vs alpha are provided in the workbook (2 individual and 1 superimposed)

## Findings

### Finding #1

A regression model with price as the target variable seems to be impractical due to its very large MSE.

### Finding #2

Use of ‘log\_price’ brought down the error to single digit, thus making the models realistic in terms of prediction of the car price. To get to the car price, one needs to take inverse log function or e^(predicted log\_price).

### Finding #3

Model #8 uses a total of 94 features and has MSE = 1.668658711858038. Out of these 94 features, the statistics and p\_value analysis both provide the same 7 features that are statistically most significant. MSE = 1.668658711858038 (Refer to comments in “Module 11 Workbook.ipynb”. Note that these 7 features do not contain any of the onehotencoder type features.

Model #9 uses 7 most significant features out of the 94 of model #8 via sequential feature selection, with n\_features\_to\_select = 7. MSE = 1.3144517741100517

This implies that model #8 is overfitted. Model #9 MSE is approximately equal to model #7 MSE.

### Finding #4

Model #10, which is a Ridge regression model that uses the same 94 features as model #8, yields a lower training MSE and test MSE when alpha = 10^(-5). Apparrently, the Ridge model with best\_alpha seems to be the best model of all the 10 models.

### Finding #5

As stated in the deployment section of prompt\_II.ipynb, the most significant features for determining car price are cylinders, year, odometer, state and condition.