Predicting new car pricing

Michael Harrington
Springboard - Capstone 2

How accurately can we predict new car prices?

Introduction

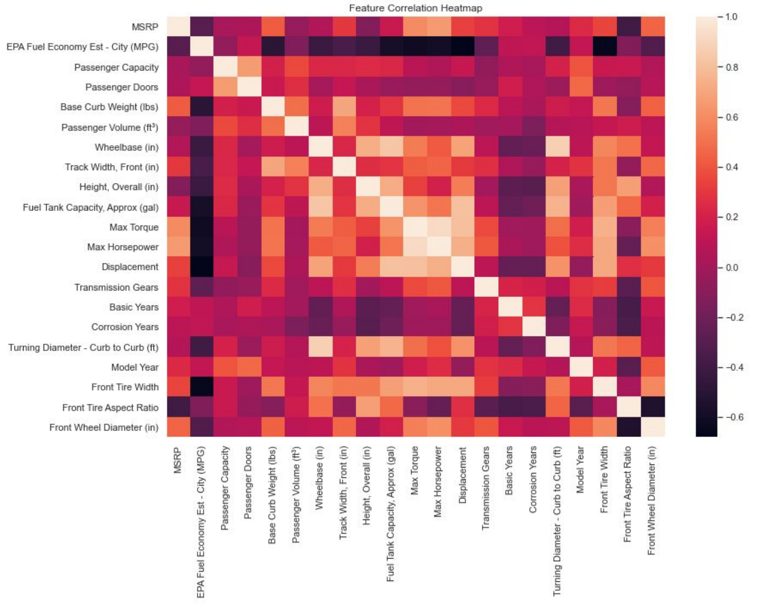
- The aim of this project was to create a regression machine learning model to predict the original price (MSRP) of a new car
- Potentially useful for a variety of companies/clients in the automotive and related fields
    - Car manufacturer - Better position their future car MSRP or project price of competition's model
    - Investor/Business Analyst - Knowing car pricing may impact a company's stock evaluation/projection

Dataset

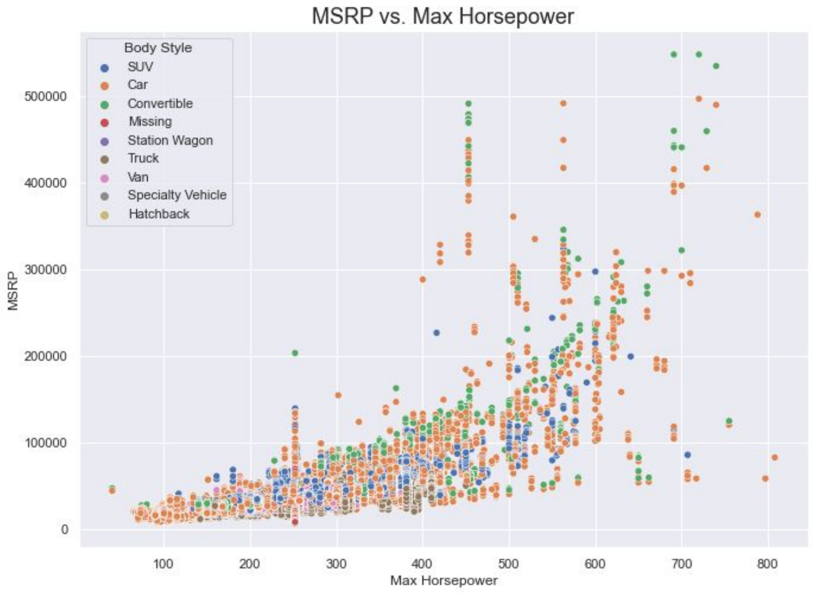
- The dataset used was found on kaggle.com can be examined here - New car prices
- Originally consisted of 32316 cars with 57 features
- Contained variety of new car models from 1991 to 2019
- Dataset isn't evenly distributed - Contains more cars in later years, which makes it more relevant in the case of projecting to the future
- Required a lot of cleaning and feature selection/extraction

Correlation Heatmap of Numeric Features

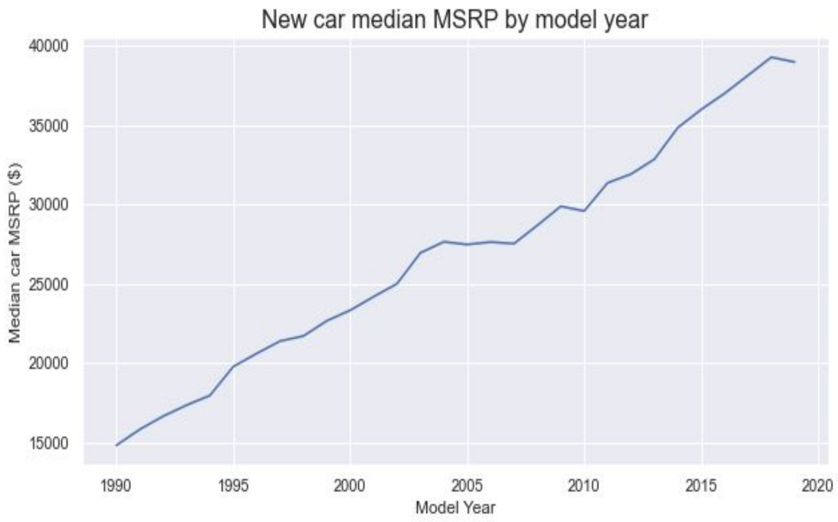
- Target feature (MSRP) shows strong correlation with variety of features including Max Horsepower, Max Torque, Base Curb Weight etc.
- Of note is the negative correlation with Fuel Economy
- Surprised Model Year isn't even more positively correlated



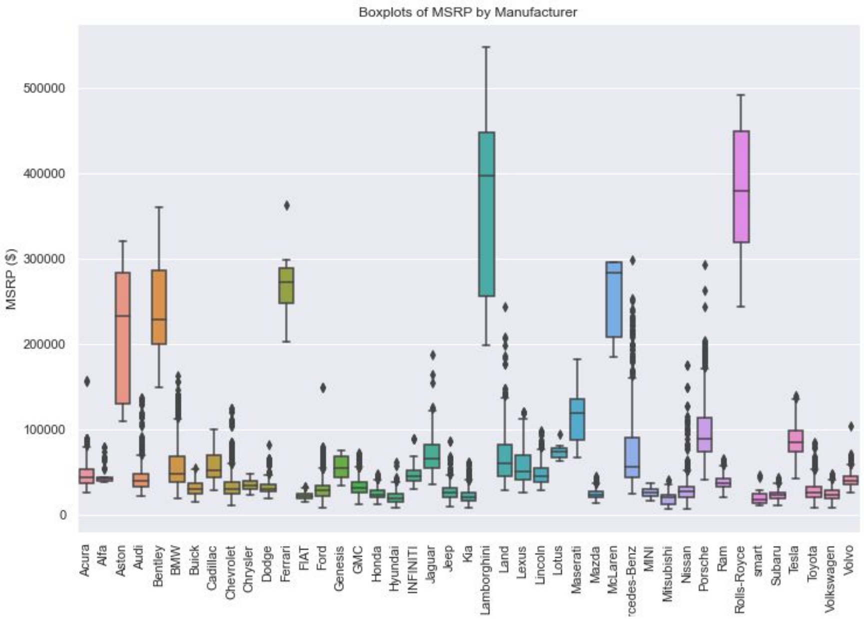
- MSRP vs. Max Horsepower
- Car and Convertible body styles dominate among outliers
- The generally strong positive correlation is evident
- The vertical lines at certain horsepower points are interesting (may represent shared engines or values targeted for marketing purposes)



- Expect Model Year to be an important feature in modeling
- The mostly consistent upward trend in median MSRP by year is clear
  - More than doubled in approx. 30 years
- The leveling off from about 2004-2007 is interesting to note
- Important to remember dataset skewed quite heavily towards more modern years

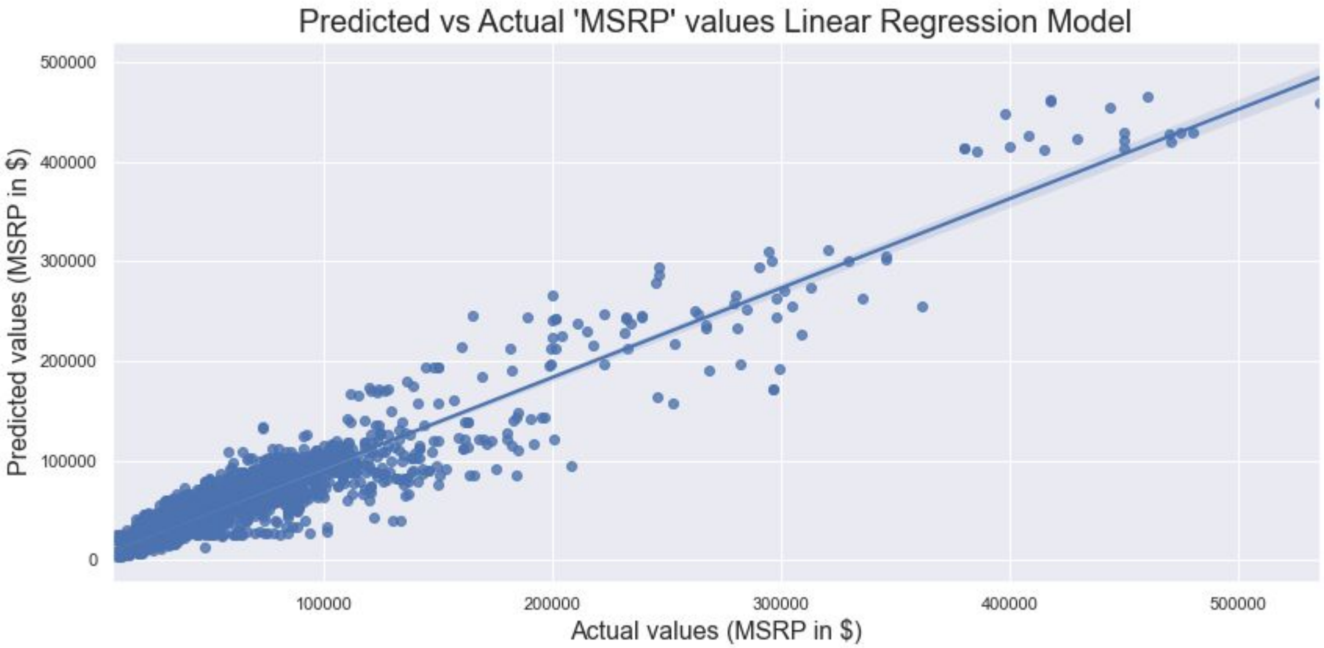


- Expect manufacturer to be key feature
- The plot highlights most manufacturers clustered below 50,000 with some outliers
- A few luxury brands display much more expensive cars as well as higher variance



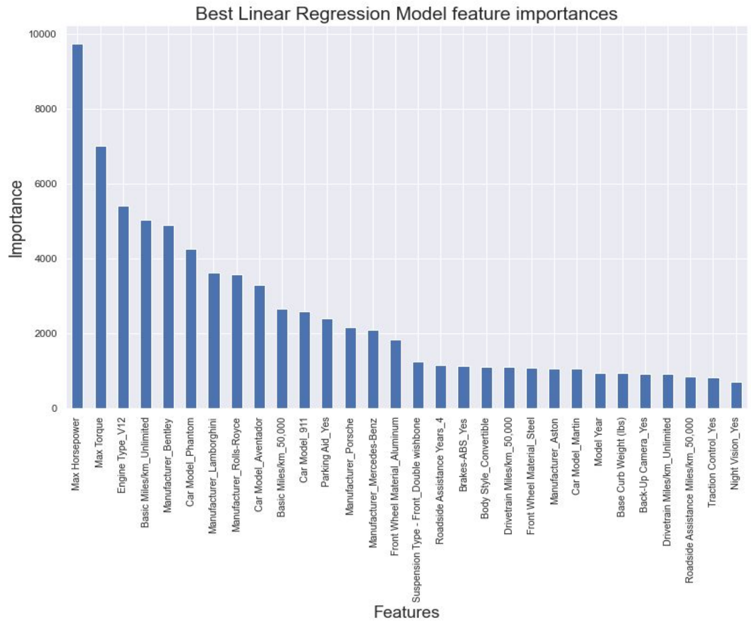
Linear Regression Model

- Wanted to try a simple model first, not expecting it to perform as well as other options
- Default model tried first, using all features - It did ok, but outliers and the large number of features seemed to give it some problems
- Tuned model with GridSearchCV - yielded selectkbest k value of k = 50 (somewhat surprised with such a high k-value)
- The best model produced decent results upon evaluation on test set : R^2 = 0.897 and a MAE = 6490.21 dollars which corresponds to percentage error of 17.15 % on average


Predicted vs Actual 'MSRP' values Linear Regression Model
500000
400000
300000
200000
100000
0
Predicted values (MSRP in $)
100000
200000
300000
400000
500000
Actual values (MSRP in $)

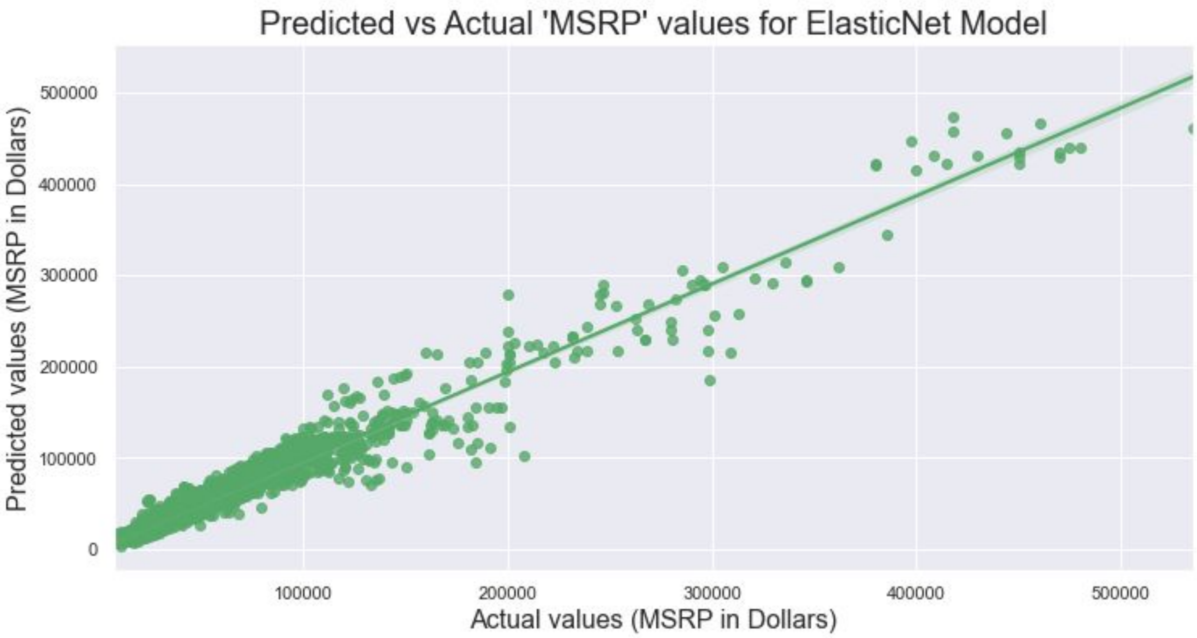
Linear Regression Model Feature Importances

- Figure displays the top 30 features for the best Linear Regression model
- The top features align with expectations (horsepower, torque, premium brands/models)
- Somewhat surprised Model Year, Base Curb Weight aren't more important based on heatmap



Elastic Net Model

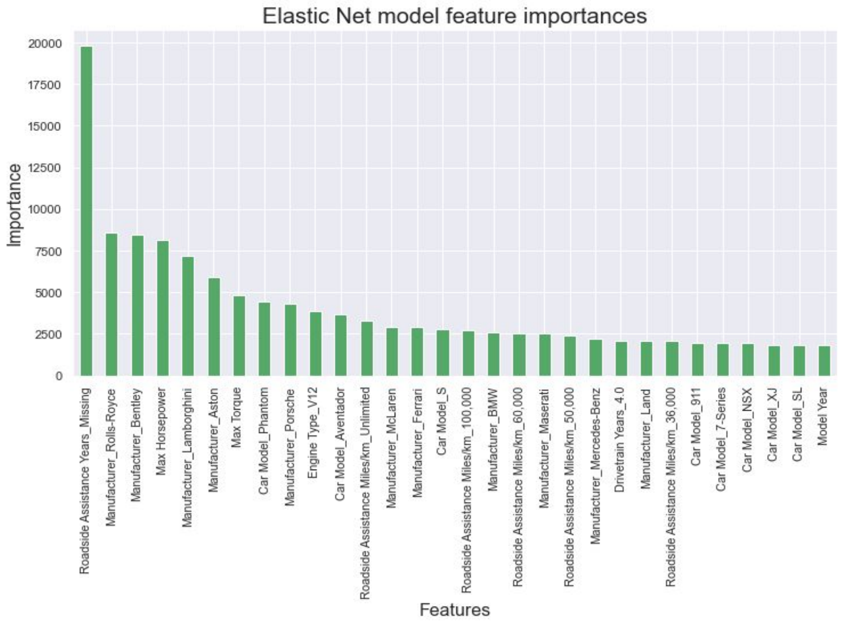
- Due to the dataset containing such high-dimensionality (large number of features), I decided to try a regularized regression model next
- Chose Elastic Net model to find the ideal form of regularization for the dataset
- Model was iterated on and hyperparameter tuning was done to find best L1 ratio
  - The tuning yielded a best L1_ratio = 1 (which is equivalent to a Lasso model)
- Best model yielded the following performance metrics: R^2 = 0.958, MAE = 3630.07 which is an average error of 9.59%
- Clear improvement over LR models



Predicted vs Actual 'MSRP' values for ElasticNet Model
Predicted values (MSRP in Dollars)
500000
400000
300000
200000
100000
0
100000
200000
300000
400000
500000
Actual values (MSRP in Dollars)

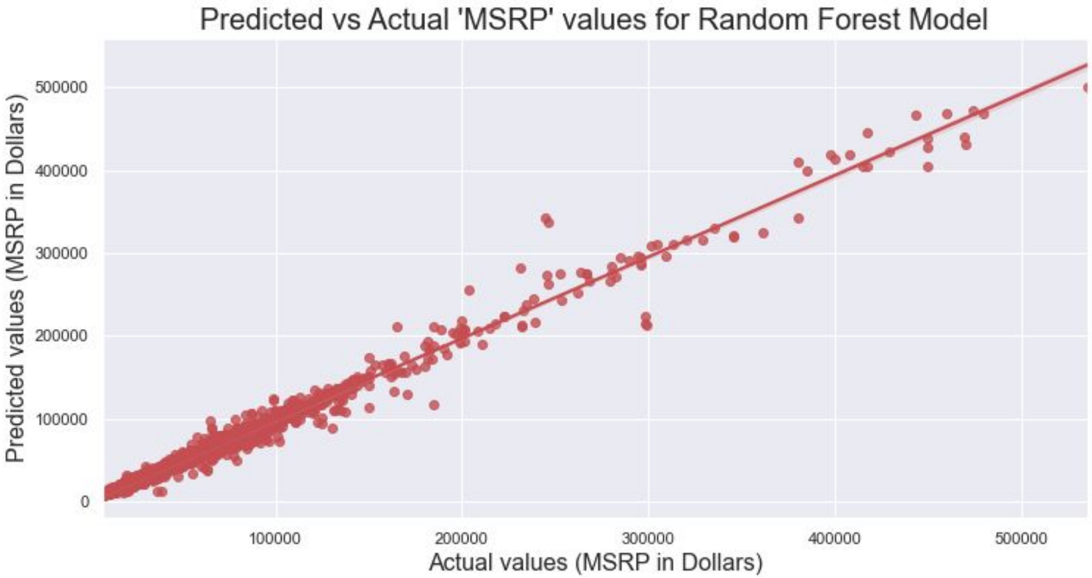
Elastic Net Model Feature Importances

- Difficult to fully grasp why Roadside Asst. was the dominant feature - Something that could be worth a deeper examination
- Beyond that it unsurprisingly heavily weighed certain manufacturers and Max Horsepower and Torque (similar to LR model)
- Model generally seemed to weigh Roadside Assistance features quite highly



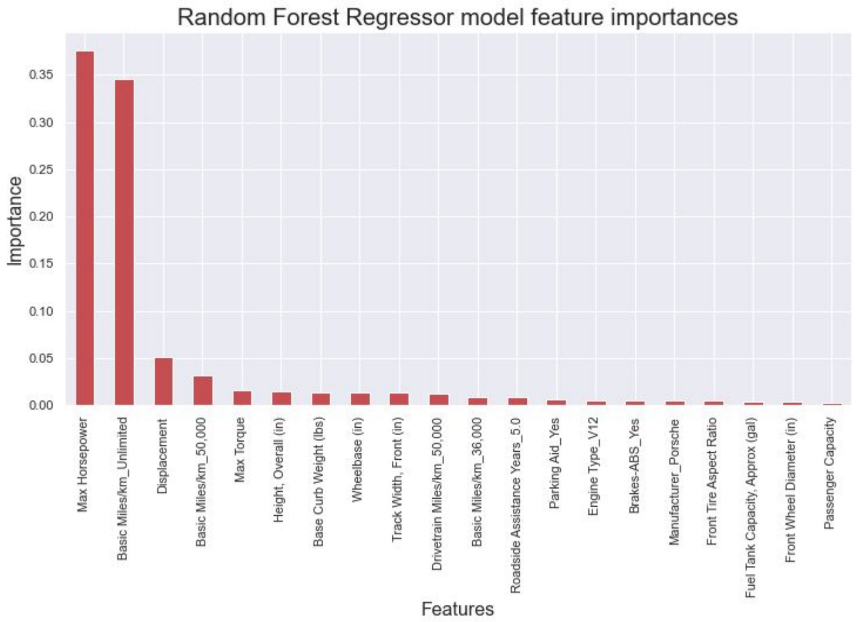
Random Forest Regression Model

- Last Algorithm chosen due to it being versatile, highly tunable, and often being extremely effective in regression tasks
- A default model was tried first, followed by a few iterations using both RandomsearchCV and GridSearchCV to attempt to tune hyperparameters (such as n_est, max_depth, max_features etc.)
- In the end the default model essentially matched the tuned models in all performance metrics, even after 5 fold cross-validation
- Best model performed quite well on test set - R^2 = 0.987, MAE = 1587.04 which translates to a 4.19 % error


Predicted vs Actual 'MSRP' values for Random Forest Model
Predicted values (MSRP in Dollars)
500000
400000
300000
200000
100000
0
100000
200000
300000
400000
500000
Actual values (MSRP in Dollars)

Random Forest Model Feature Importances

- The model clearly was dominated by two features
- A little surprised Max Torque isn't higher - I assume model found high correlation between it and Max Horsepower
- Displacement is feature I was surprised wasn't higher in the other models



Model Metrics Comparison

Model	R^2 score	MAE (in US dollars)	Relative Avg. Error (percentage)
Linear Regression	0.897	6,490.21	17.15
Elastic Net	0.958	3,630.07	9.59
Random Forest Regression	0.987	1,587.04	4.19

Random Forest Regressor is the clear choice for our final model

Summary

- Goal of the project was to create a machine learning model to predict the MSRP of new cars
    - Our target, being a continuous numeric feature, means we need to implement a regression model
- The dataset used contained approx. 32000 cars and 57 features of those cars : link to the dataset located here - New cars dataset
- Dataset was wrangled, cleaned and a few new features were added/extracted. Also a few other features were cut (too many missing values/not in usable state/contained same information/etc.)
- Dataset was explored with EDA, then pre-processed to best prepare it for modeling
- Several models were created using different algorithms and evaluated using several metrics-(R^2, MAE, percentage error)
    - Linear Regression
    - Elastic Net
    - Random Forest Regression
- The final model chosen was the best Random Forest Regression model - it performed better than the other models in every metric, exhibited low variance, and was interpretable

Limitations and Further Recommendations

- Dataset limitations
  - Contains no cars newer than 2019 models
  - Having actual sales data - (would be huge for modeling and seeing effectiveness on pricing or the changes over time etc.)
- Be more efficient with time wrangling and cleaning data
- Trying different imputation techniques and standardizations within pipelines
- Models could probably have been more efficient with further dimensionality reduction - especially in regards to Linear Regression
- Given enough time would be interesting to try further algorithms and compare results - (particularly SVM, Gradient Boosting and other ensemble methods)