

**Cohort 13**

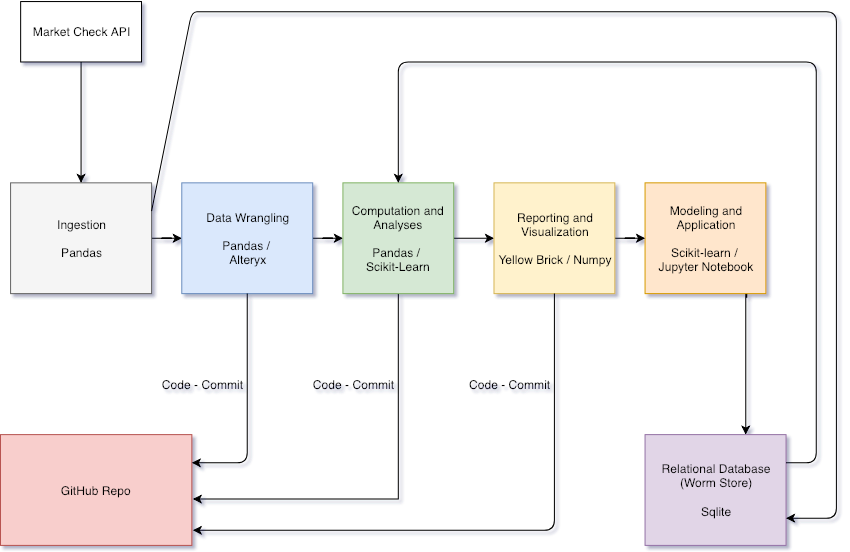
**Klunker Blue Book Team**

**Namrata Sharma, Grace Ye, Dante Bellins, Daniel Palumbo**

**Introduction:**

An estimated 39.5 million used vehicles will be sold in 2018. This continues a trend of explosive growth in used vehicle sales, and while this growth has recently slowed by a small amount this overall trend of growth is set to continue well beyond this year.[[1]](#footnote-0) With so many used vehicles on the market, we wanted to see how we could predict their listing prices using machine learning techniques. To do this, we will try to examine how the different features of a used vehicle appears to affect its price, with special attention given to the vehicle’s geographic location.

**Project Model**:

****

**Data Sources:**

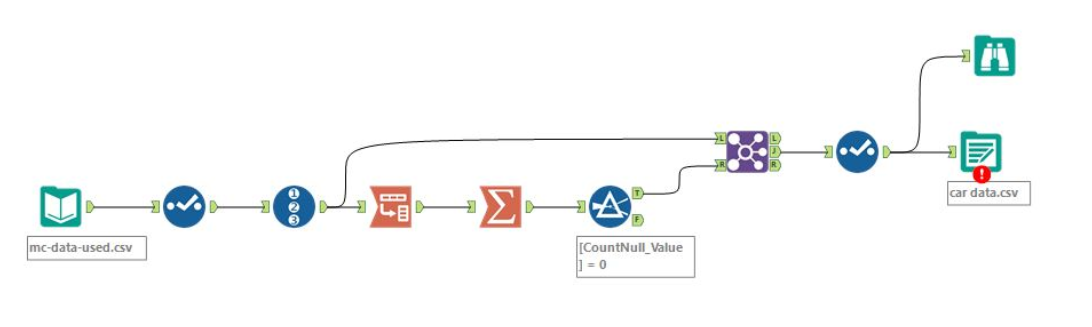
Using the Market Check API[[2]](#footnote-1), we requested a data set of their inventory data containing all currently listed used vehicles at that date. This data included used vehicles available across the entire continental United States along with a large number of features, such as their listed price. This raw data set was a simple .csv file containing a staggering **6,775,054** instances along with **38** different features, weighing in at a cumbersome 27GB. Simply reading a data set of the size proved to be a challenge. We identified and quickly dropped two string features that were overloaded with text in order to reduce the data set to a manageable size.

However, this wasn’t all the data that we required. Our vehicle data set had a feature for the zip code, but contained no further regional information. To fill in these gaps, we acquired a free data set containing further geographic information for every zip code[[3]](#footnote-2), such as state and county, as well as another small data set from Wikipedia based off of official U.S. Census Bureau data containing individual and family income information for every county in the country.[[4]](#footnote-3)

Later during the project, we returned to the Market Check API and pulled three more days worth of current used vehicle listings. However, after we examined the new data, we decided not to include it in our final analysis for a number of reasons including a large number of null values present and many duplicate instances.

**Data Wrangling:**

Our starting data set was a monster. At 27GB, it proved so large we had a lot of trouble simply loading the file. After trying a couple different approaches, including loading it into SQLite, we ended up pulling the file into Alteryx because it was able to handle the file and allowed us to look at its features notice that it contained two very large text features containing concatenated text storing various vehicle options and features. These two features were responsible for the majority of the file’s girth, and were immediately removed.



The above figure shows this Alteryx workflow, where we were able to:

* Drop the “Option text” and “Feature text” columns
* Drop columns that had only one value in them such as “seller type”
* Drop rows and columns containing *only* NULL values
* Alter NULL values for the “is\_certified” feature to “0”
* Truncate zip code to 5 digits

We named this new data set **car\_data\_clean.csv**, containing **3,509,809** instances with **30** features. A short while later, we became more familiar with Python and decided instead of including Alteryx in our pipeline, we would keep things consistent, and repeated these steps in Python inside a pandas dataframe, where we also performed the rest of the wrangling.

The county demographics and zip code data sets were also imported into pandas dataframes as **county\_demos.csv** and **zip\_code\_database.csv**. Once there, a number of zip codes had to be dropped due to being defunct or outside our scope, such as those assigned to military bases, and we had to engineer a new feature for county\_demos.csv containing the state’s two letter abbreviation for every county. With that, we were able to use the state abbreviation feature to inner join the county and zip code data sets together, and then in turn use the zip code feature to inner join our main car data set with this demographic information. Afterwards, we quickly did a few more alterations including:

* Renaming the features to something a little less awkward (e.g. “price\_fs” became “price”, “make\_ss” became “make”, and so on)
* Dropping a few redundant columns that snuck in our join or provided unhelpful information
* Reformatting income feature values to be integers, removing the $ and commas
* Dropping duplicates
* Engineering a new feature, “AGE” containing the age of each car, using its model year.

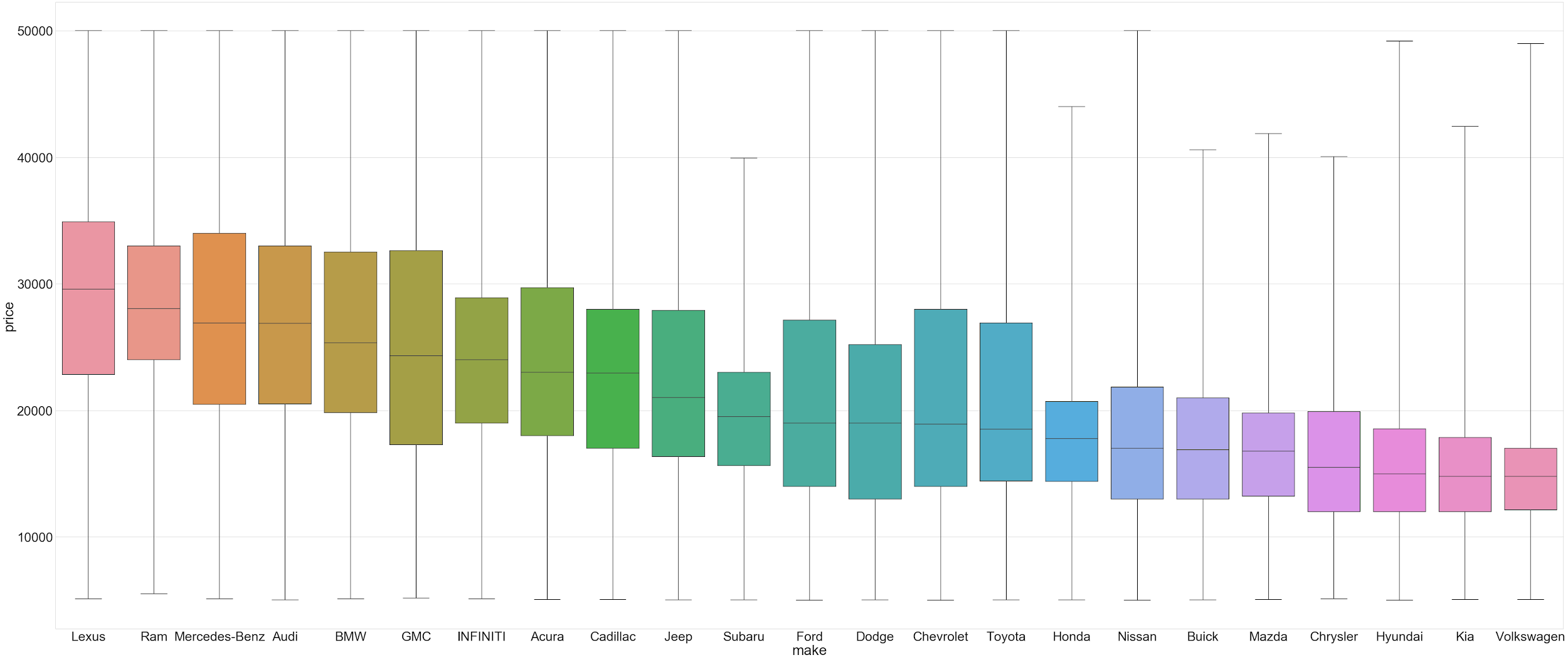
However, it is worth noting that while in the process of dropping duplicates, we discovered that many times the same car would be listed in multiple states, after consideration, we decided to not consider these duplicates and keep them as separate instances as much of our analysis would focus on vehicle listings by location, making these listings all valuable data, especially since in some cases the listing price varied by location. This, after a few iterations, resulted in the main data set we will be using in our visualization and modeling, the slightly cumbersomely named **car\_data\_clean\_complete\_v4.csv**, now containing **1,673,470** instances and **29** features.

To start with, and to get some practice, we looked at the data volume per vehicle make and extracted the top 10 vehicle makes by volume for our initial data visualization and modeling. But once became comfortable with the data, we went back and decided to instead cut any car make with less than 15,000 instances in our data, resulting in us using the **top 23 vehicle makes** which is seen in our visualizations and modeling below. We removed any price outliers, which we defined as any used vehicles listed for less than $5,000 or more than $50,000.

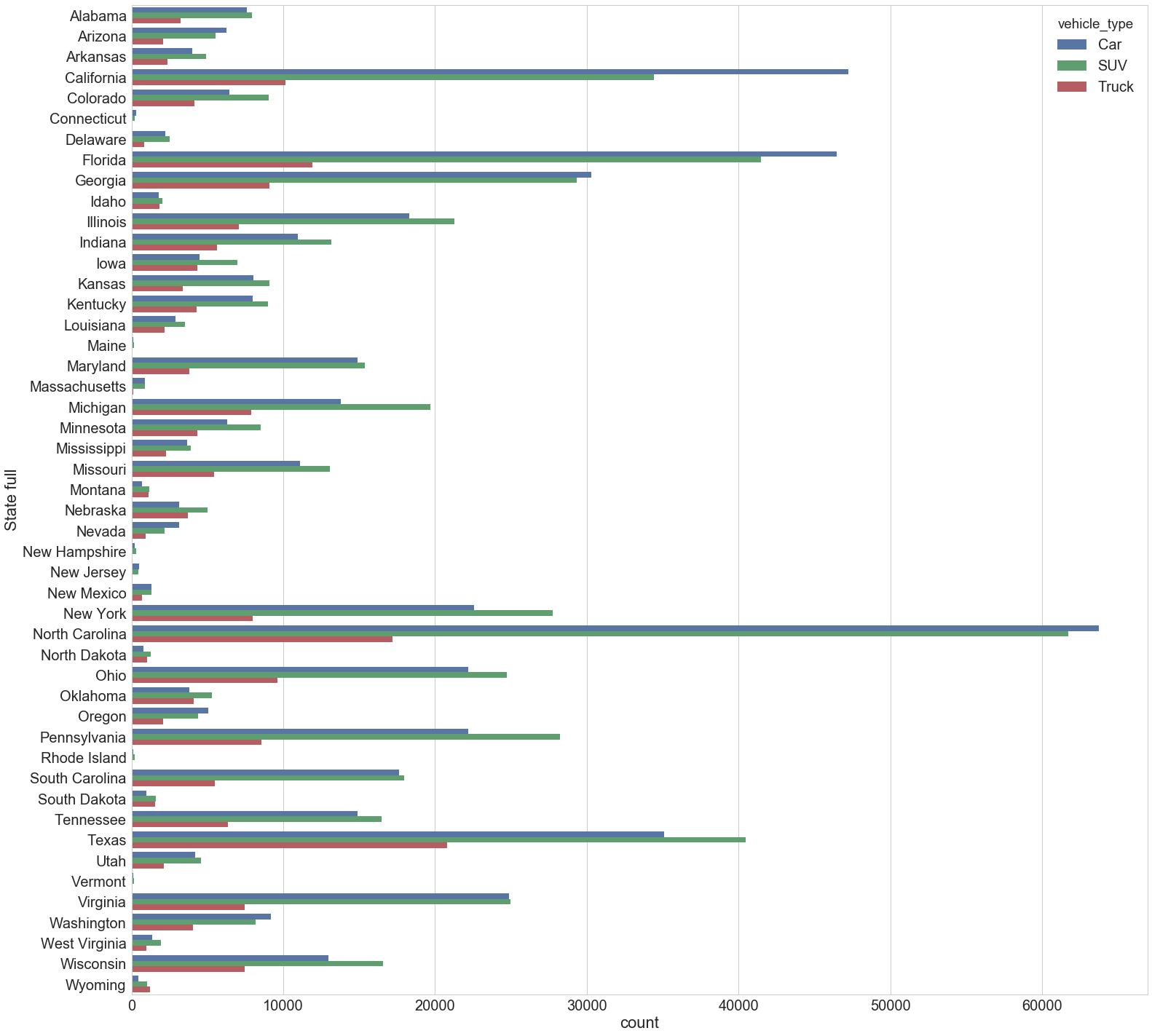
As we neared the end of the project, we saw more interesting questions. These questions prompted the addition of regions. These regions included typography, weather conditions (typical for a certain area), and historical economic significance. Because of the thorough cleaning performed above, we were able to add multiple forms of regions to the **car\_data\_clean\_complete\_v4.csv** without losing many instances. Once the inner join on abbreviated state names was complete, we had our amended base dataset, **car\_data\_clean\_complete\_v5\_ext.csv**, which contained **1,673,412** instances and **49** features. This latest version of the dataset was only used at the very end. If our group had more time, we would have used this richer dataset to provide more insight.

**Data Visualization:**

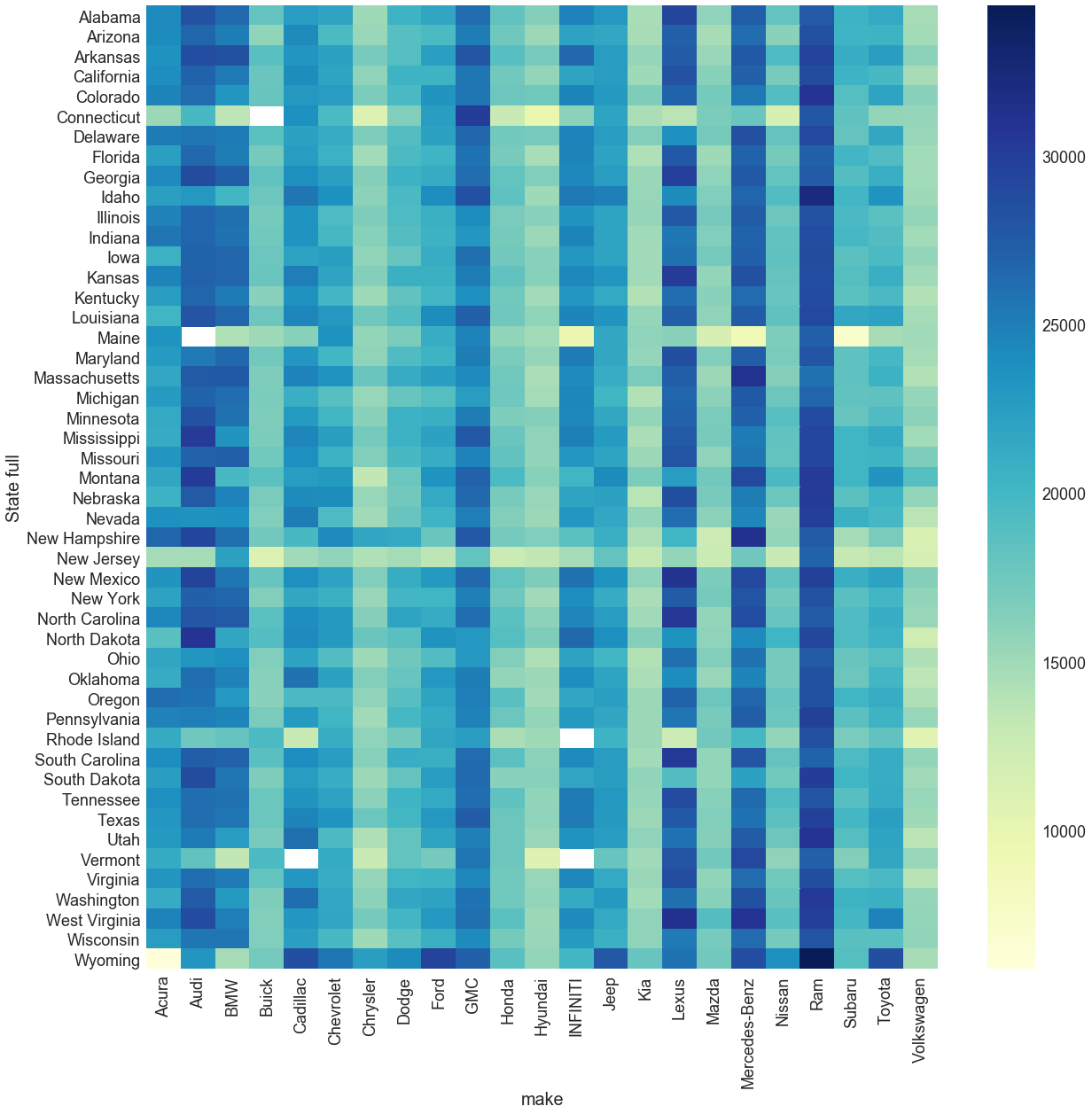
For visualizations, we employed a combination of pandas, Seaborn, and Plotly. First, our team started by tallying [the number of listings by vehicle make](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/MakeCount.png), which, as the link shows, Chevrolet and Ford are by a wide margin the most popular vehicle makes in the country, leading to a steep drown off before evening out to a steadier decline.



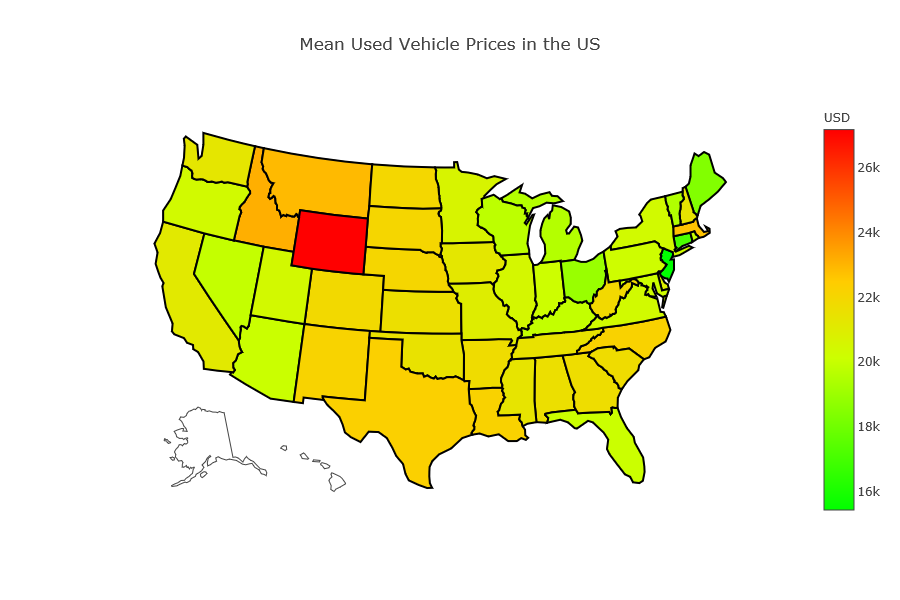
We next added to this by creating a boxplot to demonstrate the price range of each make. This proved interesting as it showed that all of the lower limits of each make’s listings butt up against our imposed lower than $5,000 cut off point, and *most* of them are also against our upper $50,000 upper limit point. Even so, each make’s medians and quartiles are unique, and display clearly the price range each maker tends to offer, with cheaper makes generally having less price variance and becoming more similar in shape.



Next, we looked at the listings by state, and this provided our first bits of insight into the limits of our data. Breaking it down, instances range from over 181k in North Carolina, down to 340 in Vermont. This makes accurate comparisons between them quite tricky. Even inside North Carolina, the most represented state, its records by county range from over 53k in Mecklenburg County to a grand total of 3 in Cherokee County. While we initially hoped to examine regional trends at a county level, these gaps of data simply wouldn’t support that. For this reason, we had to abandon the idea of county levels analysis and settle for looking at the graphs and predictions based on the state level, as it's the closest we can get to equal and substantive data to compare with, while still allowing for regional trends to emerge.



This heatmap we then created added another dimension to our data, letting us compare the price of each make against every state. Here we can see how the more expensive brands retain their command of higher prices across the states, but also where certain cheap North Eastern states that are thin on listings still drag that price down below their national average. It’s easy to notice also that there are a handful of white spots standing out on the map. Unfortunately, they do not indicate that you can, for example, get an Audi for dirt cheap in Maine, but instead that there is no data of Audis listed in Maine. It again illustrates that despite our data set’s sizable 1.6 million instances, gaps in it remain.



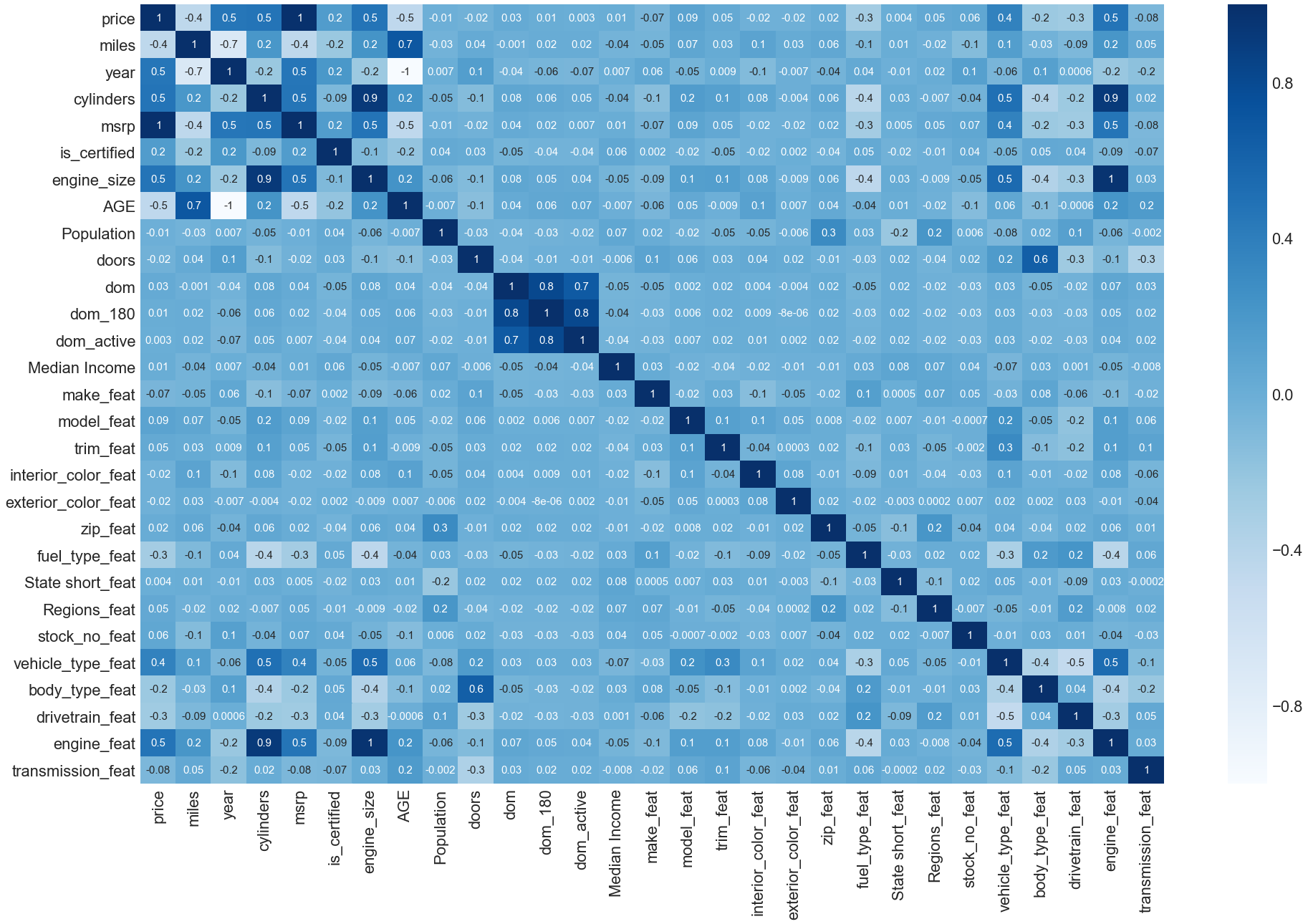
But to truly examine statistics across the US, we have to be able to picture the US. This gave us some trouble until we found the choropleth map features in Plotly, with which we were able to easily see trends across the country. In particular, looking at the price across states, a few places instantly catch the eye, so we decided to start by understanding why the outliers stood out.

Wyoming stands out starkly as the most expensive state in the nation, being 27.8% more expensive than the national average ($22k national vs $28k WY). But why? It didn’t seem a coincidence that the least populated state in the nation is also the most expensive used vehicle market, but while this is certainly related, the answer mostly came from two other facts gleaned from examining the data in different ways. The first being that the previous breakdown of different vehicle types listed in each state revealed that Wyoming in the only state in the nation with more trucks for sale than cars or SUVs. And, looking at the [average price for each vehicle type](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/VTypePricepng.png) quickly reveals that the average price for a used truck is considerably higher than cars or SUVs, explaining why Wyoming stands out in the nation. However, it should be noted that even with this consideration, the price for trucks and SUVs in WY still sits on the higher end, while cars are decidedly average, which also helps signify how the demand for different vehicles in rural areas varies.

Meanwhile, prices listed in New Jersey average are cheaper than anywhere else in the nation, 21.2% lower than the national average. This especially stands out against neighboring states. But coming to understand why proved a bit tricky. To see if NJ prices were universally cheap, we took several of the most commonly listed cars throughout the entire US. The single most common being [the 2015 Chevy Silverado 150](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/MeanSilverado.png), with over 15,000 listings across the country. This reveals that at an average of $29.3k for a Silverado in New Jersey putting it solidly in the middle of the pack. Examining other popular models did did show them slightly on the low side for NJ, enough to start to paint the picture of the state being a very cheap market. In order to go deeper into this, we then looked to the [mean age](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/MeanAge.png) and [mileage](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/MeanMileage.png) for all vehicles in the country. And in both of these categories New Jersey sits at or near the top of the pack. Clearly, used vehicles in NJ are older, more worn, and cheaper than elsewhere. And given New Jersey’s close proximity and relations with neighboring states, and the fact that stands out as a strange anomaly amid the most of the New England region, what seems to be most likely occurring to us is that most of the used vehicles instead get listed in in New York or Pennsylvania instead.

In order to understand the overall regional trends, we compared everything we’ve seen so far, such as the [mean age](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/MeanAge.png), the [mileage](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/MeanMileage.png), as well as average temperature[[5]](#footnote-4) and population of each state.[[6]](#footnote-5) When compared to the above figure of the average price the pattern became clear to us that the states with a lower, more rural population are the states which also have the highest prices, partially because of their increased demand for trucks and SUVs compared to other states. States with a more urban and centralized population, in contrast, have much lower average prices, up to $6,000 less for the same car when compared to the rural states.

**Feature Selection:**

****

To start off with, we wanted to understand how each feature, and in particular price, was was related to each other, and to do that, we use the Label Encoder function from Scikit-learn to convert our 8 categorical features into numerical values, then with that created a heatmap using the Pearson correlation coefficient. The first thing that stands out, is that MSRP is an almost exact match with price, and by looking at some of the instances, we noticed that in many cases the MSRP was indeed often the same exact number as price. Finding this to be at best unhelpful, and at worst ridiculous, we knew immediately to remove the MSRP feature from our future modeling and visualization consideration.

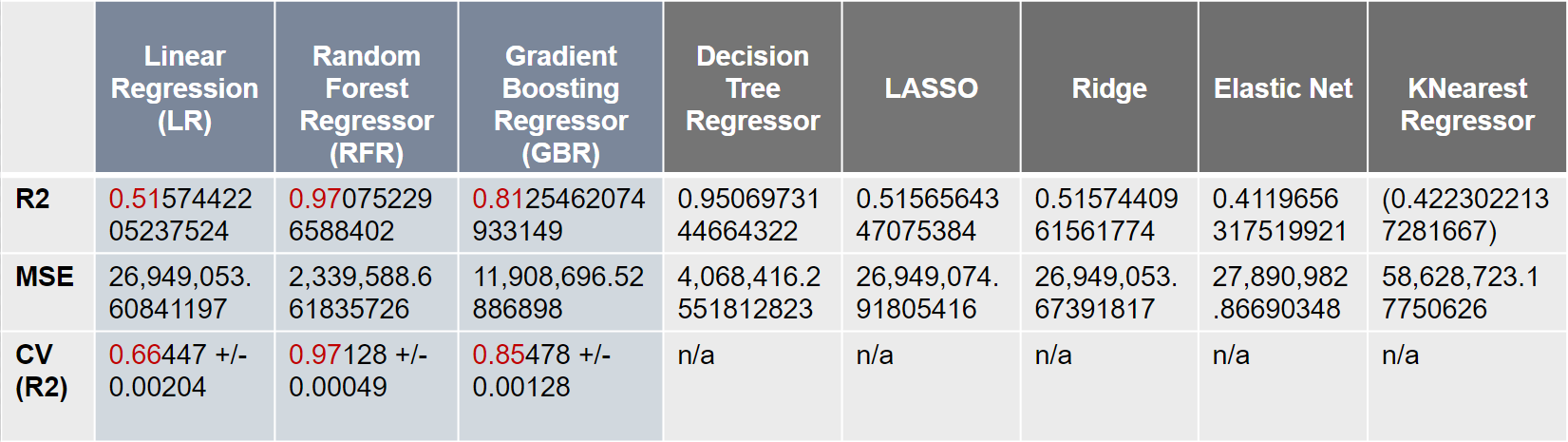
The results then show that the biggest price impactors are, unsurprisingly, the miles the cars has, its age/model year, its engine size and features, and the type of vehicle it is (car, truck, or SUV). While the lowest impactors appear to be its days on the market (“dom”), its interior and exterior color, zip code, and median income in its area. Curious why the exterior and interior color would have such a small impact, we examined the data and discovered that the majority of vehicle colors are listed using their own marketing name, such as “Silver Ice Metallic” or “Crystal White Pearl Mica,” making vehicles that would to the layman be the same color qualify as completely different in our data. Unfortunately, the vast amount of different names used in our data set made converting them into a unified standard unfeasible given the time limits of our project, but we decided to keep them for the moment.

We then identified the features that would become redundant in our modeling such as “year” compared to “AGE”, “zip” compared to “state”, and dropped them, to use a total of 25 features as the “X” in our model, with price of course being the “y.”

Later in the project we returned to reexamine our feature selection using new information. In this ‘phase two’ test, we ran feature importance functions using models such as [LASSO](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/LassoFeatures.png) and [Ridge](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/RidgeFeatures.png) to generate graphs displaying the relative coefficient magnitude of each feature. Using this, we were able to confident reduce the number of features down to the twelve most impactful ones, and evaluate their importance across even more models, like [Linear Regression](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/LRFeatures.png), [Random Forest Regressor](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/RFRFeatures.png), and [Gradient Boosting Regressor](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/GBRFeatures.png).

**Modeling:**

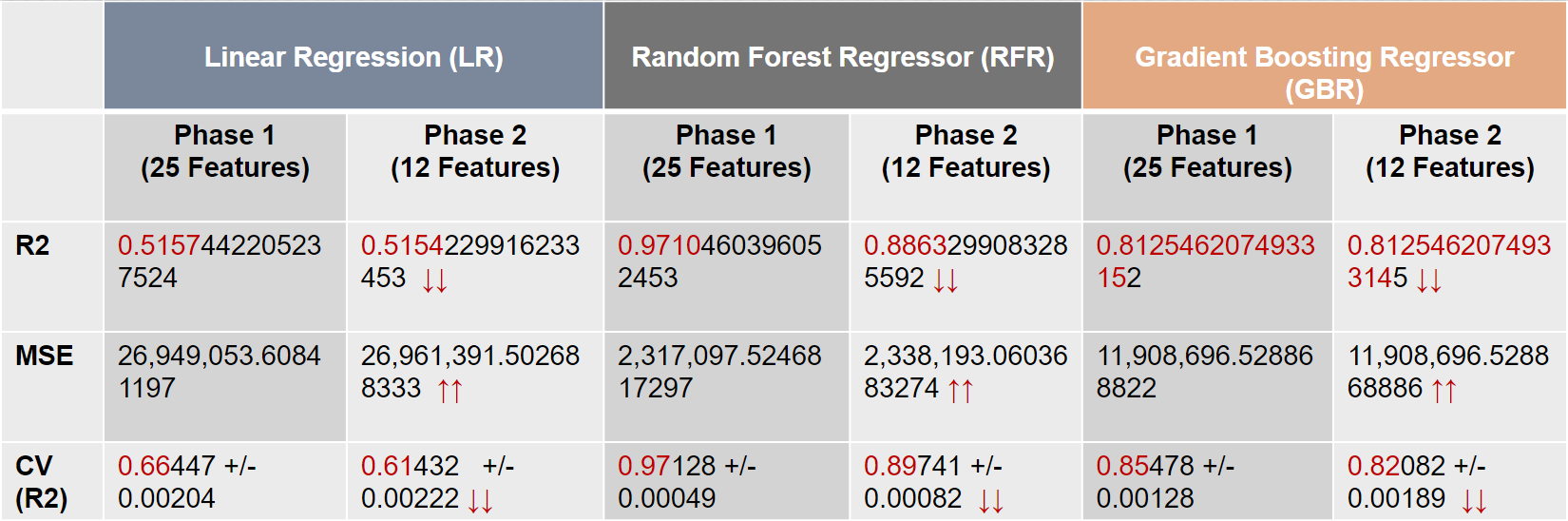
For our initial modeling, we chose eight different models from four different model families that we were the most familiar with to try and fit our data, with 25 features, onto. From the Linear model family, we used **Linear Regression**, **LASSO**, **Ridge Regression**, and **Elastic Net**. From the Ensemble Methods family of models we used **Random Forest Regressor** and **Gradient Boosting Regressor**. And finally, in order to get a more outside perspective, we used **Decision Tree Regressor** from the Decision Tree family and **K-Nearest Neighbor Regressor** from the Nearest Neighbors family. Another model we tried was Support Vector Regression, but unfortunately none of our attempts to fit it worked, as the model caused memory errors for everyone in our group. But out of these eight models that we were able to fit, our results varied significantly.



Elastic Net and K-Nearest Neighbors Regressor stand out in our results, with a dismal sub 0.5 R2 score. Unfortunately, Linear Regression, LASSO, and Ridge Regression don’t fare much better, with all three displaying very similar R2 scores of 0.515. But RFR, GBR, and Decision Tree, show some promise, with 0.97, 0.81, and 0.95 scores, respectively. In order to get a breadth of different options, we chose **Linear Regression**, **Random Forest**, and **Gradient Boosting** to explore in more depth.

Our next step was to then use Yellowbrick to create a Residuals Plot and Cross Validation Score graph for [**LR**](https://raw.githubusercontent.com/georgetown-analytics/Klunker-Blue-Book/master/Final%20Presentation/KBB%20Images/LRResults.png), [**RFR**](https://raw.githubusercontent.com/georgetown-analytics/Klunker-Blue-Book/master/Final%20Presentation/KBB%20Images/RFRResults.png), and [**GBR**](https://raw.githubusercontent.com/georgetown-analytics/Klunker-Blue-Book/master/Final%20Presentation/KBB%20Images/GBRResults.png). With these, we observed that all three models are fairly well clustered around the mean 0, with RFR being the most centered. Although all three share a small but interesting trends upwards as price increases, indicating to us that the more expensive the used car is, the more its categorical features, like its make and model, are valued over the integer features like age and mileage. RFR was also notable for having the highest R2 score along with the smallest R2 deviation across our cross validation tests and the lowest mean squared error (which is also why we chose it over the similar Decision Tree).

But we were concerned about RFR’s suspiciously high R2 score, and, wanting to see what changes reducing the model’s dimensionality would bring, we returned to examining the features for our ‘phase 2’ round of selection. As outlined there, we ran feature importance functions on several models and ended up reducing the features from 25 down to the 12 most consistently important features. Then we ran our tests again and compared.



With this new selection our models are far less complex, run much faster, and focus on the features that matter most. We also see across the board decreases in every R2 and CV score, as well as an increase to every mean squared error. Clearly this feature reduction was not working our for our LR or GBR models. But what about the RFR model?

After days of discussion and debate among our group, with a lot of back and forth, but ultimately we decided that our concerns of overfitting on the RFR with 25 features were too great to ignore, and that the 12 feature version of our RFR model stood out, with its high but not outrageous R2 and CV scores, low mean squared error, and a consistent [prediction error plot.](https://github.com/georgetown-analytics/Klunker-Blue-Book/blob/master/Final%20Presentation/KBB%20Images/RFRPredError.png) All of these factors lead us to select **Random Forest Regressor** **with 12 selected features as our model of choice.**

**Data Product:**

With the Random Forest Regressor being our model of choice, we moved to building an application to display our ability to predict car prices. Our use of frameworks, Django and Flask, over Jupyter Widgets came down to our ambitious goal to produce a full fledged application. We were only partially successful.

The application would help customers in the midst of their daunting car buying process. The most valuable insight we could provide would be a price prediction on a user selected vehicle. The other insights we sought to display were car options and pricing based on varying conditions. Some of theses conditions included: region specific pricing, best price nationwide, and closely related car models that might be worth a test drive.

As cited by all of our professors, we used the majority of the time wrangling and creating visualizations to get a firm understanding of our dataset. While we remained eager to finish the application by the end of the Cohort, the task of solidifying a particular product and user experience was difficult. Our ambitious goal proved to be more than we could produce.

First, we looked into the development aspects of both Flask and Django. Django seemed to be the more robust framework, so we proceeded with it. While we could stand up a website fairly easily, we did not find the connecting of our pickled regressor to a suitable user interface to be easy. When we decided to move from Django to Flask, for perceived relative ease of use. The same issue arose. We could get a nice home and about page together, but html and Heroku proved difficult to troubleshoot.

With all that being said we do have the code to produce a car price. There are no bells or whistles, but the price does represent a reasonable price for a selected car. The inputs include, Model, Make, Year, State, and Zip. Since our time was limited, we did not have the opportunity to polish our predicted price, so our best data products became the visualizations.

Our goal was to empower the customer with the ability to make an informed decision on a heavily subjective topic. The following visualizations would inform the customer just enough as to not over burden, but more than enough to bargain with any veteran sales associate.

* Car Prices by State
* Mean/Median Make Prices by State
* Mean/Median Make Prices by Region
* Mean/Median Body Type Prices by State and Region
* Popularity of Vehicle Make by State and Region

In the future, we could produce deeper more interactive visualizations through Jupyter Widgets and other plotting packages. Also, we could produce a consistently updating Top 10 for each aspect of the vehicle purchasing decision:

* Best Cargo Room
* Fuel Economy
* Biggest Engine per Dollar
* Hatchback vs Small Sedans
* SUV vs CUV vs Hatchback
* Etc.

As we worked through the data science pipeline, we found that the fidelity of the data was the most important, but without visualizations we would have been lost. We believe that to be true for the customers as well.

**Conclusion:**

Our starting questions were all about price prediction, we wanted to know how well a used vehicle’s price could be predicted with machine learning, and how its own features, as well as more external feature such as its location, could affect its price. And after going through the modeling pipeline, our conclusions are at times both in line with, and unexpected, compared to our initial intuitions.

Our data has shown that most often, the biggest predictors of a vehicle’s price is its age and mileage, more so than its location or other attributes such as engine size. However, we were able to clearly see that the state the car was sold in still mattered considerably, swinging the price of the most popular cars by as much as 11% above or below the national average. The lower populated and more rural MidWestern states were shown to command a distinctly higher premium over the states with a more dense and urban population, despite the MidWestern vehicles also leading them in mileage and age as well.

For modeling, **Random Forest Regressor** using 12 selected features proved to be our winner, sporting one of the highest better R2 and CV scores among our models, while not being outrageous to fall into the overfitting trap, as well as the lowest mean squared error.

In the future, given the chance to work on this project further and expand upon it, we would first be eager to explore our model in more depth and see how we could tweak it to get even more consistent and useful results. Beyond that, our hope would be to write individual models for each vehicle type and utilize **Bagging Regressor** to aggregate their individual predictions, either by voting or average, to form a final prediction that we could then use as a comparison to our current RFR model. We would also try to flesh out our current features by normalizing the exterior and interior color options and add more features that are currently lacking, such as the used vehicles condition and interior components. And with those first steps, to take what we’ve learned and expand our data set far beyond its current snapshot form to integrate data scraped throughout the year in order to research seasonal price fluctuations that may exist.

Finally, we would look to create a true pipeline that can continuously ingest and integrate new data as we refine and streamline the process to continue building and improving our model and its predictions.

**References:**

KBB’s GitHub repository on Georgetown. <https://github.com/georgetown-analytics/Klunker-Blue-Book>

Manheim, 2018 Used Car Market Report. <https://publish.manheim.com/content/dam/consulting/2018-Manheim-Used-Car-Market-Report.pdf>

Marketcheck (2018). <https://www.marketcheck.com/automotive>

Wikipedia, List of United States counties by per capita income (2018). <https://en.wikipedia.org/wiki/List_of_United_States_counties_by_per_capita_income>

United States Zip Codes (2018). <https://www.unitedstateszipcodes.org/zip-code-database/>

Current Results, Average Annual Temperature for Each U.S. State (2018). <https://www.currentresults.com/Weather/US/average-annual-state-temperatures.php>

Wikipedia, List of U.S. states by population (2018). <https://simple.wikipedia.org/wiki/List_of_U.S._states_by_population>

1. <https://publish.manheim.com/content/dam/consulting/2018-Manheim-Used-Car-Market-Report.pdf> [↑](#footnote-ref-0)
2. <https://www.marketcheck.com/automotive> [↑](#footnote-ref-1)
3. <https://www.unitedstateszipcodes.org/zip-code-database/> [↑](#footnote-ref-2)
4. <https://en.wikipedia.org/wiki/List_of_United_States_counties_by_per_capita_income> [↑](#footnote-ref-3)
5. https://www.currentresults.com/Weather/US/average-annual-state-temperatures.php [↑](#footnote-ref-4)
6. https://simple.wikipedia.org/wiki/List\_of\_U.S.\_states\_by\_population [↑](#footnote-ref-5)