Exercise 2 - SDS323

Alt Nayani and Conor McKinley

3/13/2020

(1) KNN Practice

Question: How does the relationship of mileage on price level change given different trim models? To be able to answer this question we will look specifically at the Mercedes S Class vehicles.

Let us begin by first analyzing the general relationship of mileage on price levels without any regard to specific trim models.

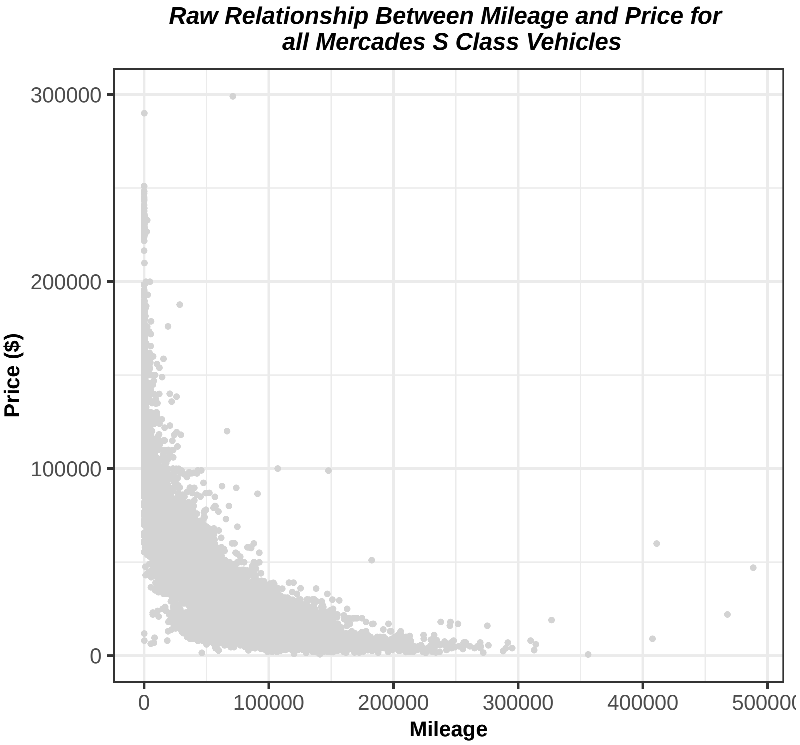
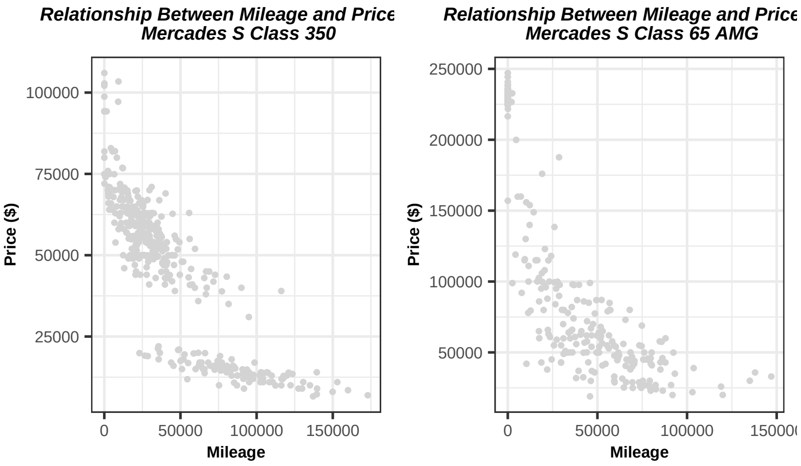


Figure 1: Data contained over 29,000 vehicles that were advertised on the secondary automobile market during 2014

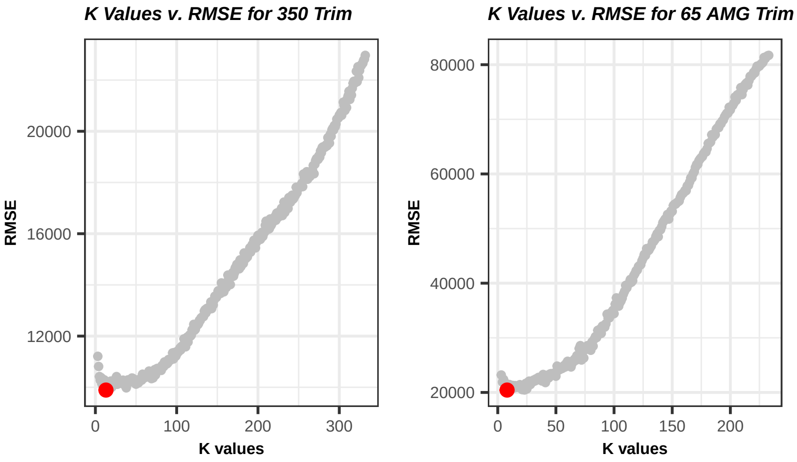
The relationship is an exponential decay relationship in which as mileage on a car increases, the price of the car drops off significantly.

Now lets analyze two specific trim levels: the 350 and 65 AMG. As seen below both trim levels follow a similar exponential decay pattern. An interesting observation for the 350 Trim is the clustering into two different groups.



To be able to quantify the impact of mileage on price, a regression is the obvious choice. Given the non-linear relationship of mileage on price and the clustering of specific values in certain trim levels, a K-nearest-neighbors regression could be used.

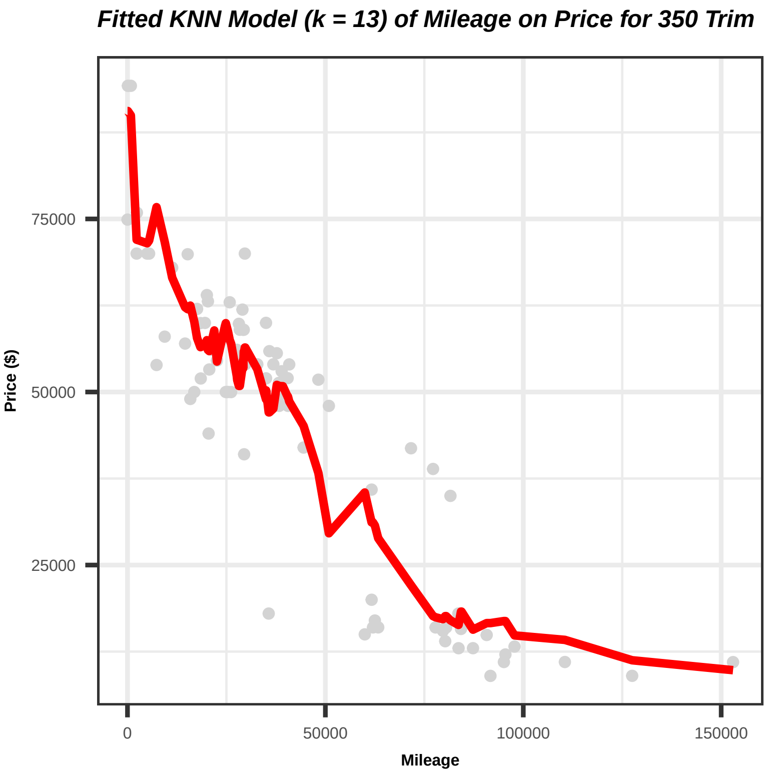
To determine which K value should be used, we should attempt to minimize the out of sample root squared mean error (RMSE). Below is a plot of K values and their associated out of sample RMSE for the 350 ad 65 AMG Trim Levels.



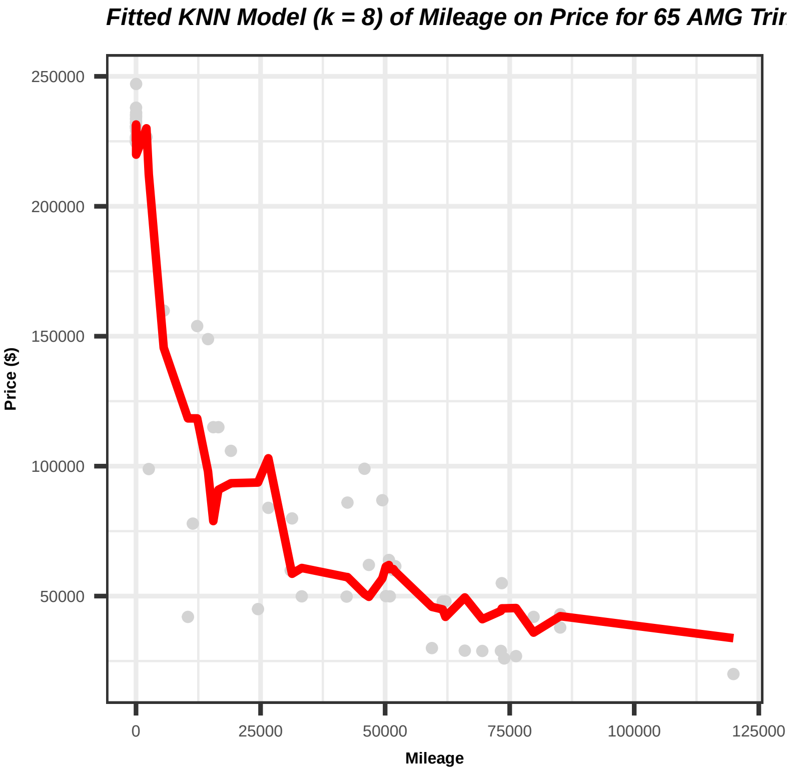
The lowest RMSE value is highlighted. For the 350 Trim a K value of 13 produced an RMSE of 9,892 and for the 65 AMG Trim a K value of 8 produced an RMSE of 20,432.

Now that we know the K value that produces the lowest error, we can run a K-nearest-neighbor model with the given K. Below are the plots after the KNN model (tested on the training data) is fitted to the testing data.

First is for the 350 Trim Level:



Then for the 65 AMG Trim Level:



Conclusion: Based on the given models above, we can now determine a price for the car knowing the mileage of the car for the 350 and 65 AMG trims. The model performs significantly better for the 350 trim than the 65 AMG (approximately 106% better). The underlying cause behind such a wide discrepancy is the clustering in the 350 Trim. Running a KNN model with clustering in the dataset provides more accurate predictions when running on the testing data, because the clusters provide relatively close predictions for the given vehicles. Furthermore, the overall data in the 350 trim seems to be more compact than the wide spread of the 65 AMG trim. In general, this would also produce a better prediction model for the 350 trim.

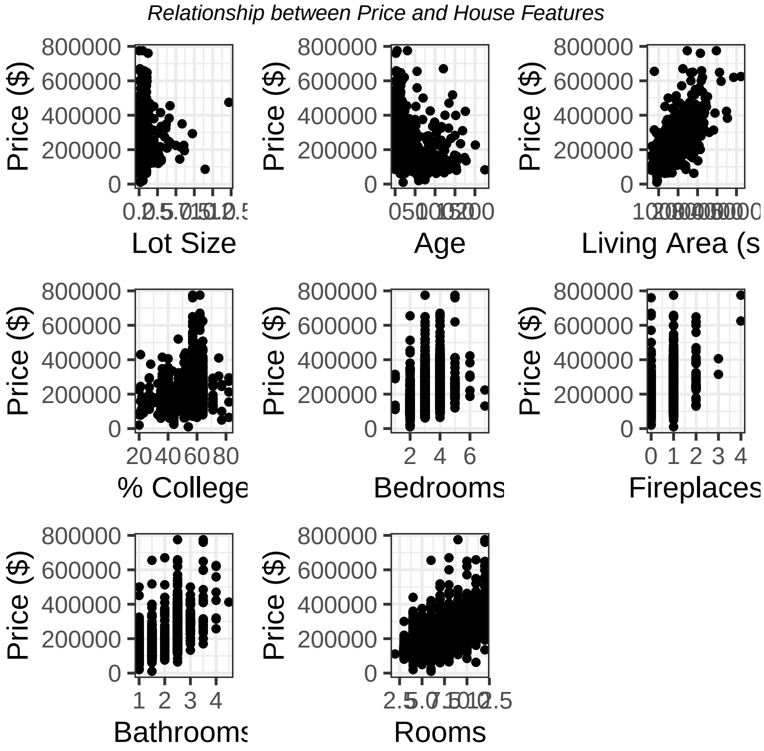
(2) Saratoga House Prices

Question: How do we best determine the fair value of a house to appropriately collect the relevant taxes?

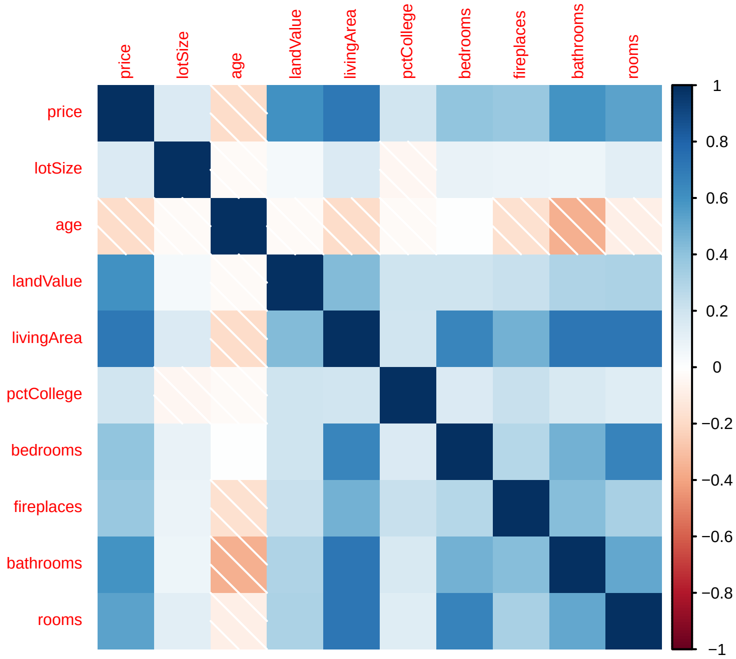
Our current model uses an array of factors such as lot size, age, living area, percent of college residents, number of bedrooms, fireplaces, bathrooms, rooms, type of heating, fuel type, and the existence of a central air system.

While our current model performs adequately, there has been discussions to attempt to create a better model since it would provide a greater accuracy of the true tax liability and thus decrease the chances of the tax department being over/under paid.

To determine the best factors to place into the model, lets first focus on the relationship between our variable of interest, price, and other quantitative features.



We can quickly visualize which features are correlated with price. However, to be able to quantify these relationships, a correlation matrix is needed.



Now to create a model based on the features that are most correlated with price. Using forward selection, a model was determined fit to best minimize error in the testing data using the correlated features. Below are the features and formula of the model generated:

```
## lm(formula = price ~ livingArea + waterfront + centralAir + bathrooms +
##     fuel + lotSize + bedrooms + rooms + newConstruction + pctCollege +
##     sewer + livingArea:centralAir + livingArea:fuel + waterfront:fuel +
##     centralAir:fuel + fuel:bedrooms + centralAir:bathrooms +
##     centralAir:newConstruction + fuel:pctCollege + waterfront:lotSize +
##     waterfront:pctCollege + centralAir:pctCollege + lotSize:pctCollege +
##     livingArea:rooms + bedrooms:rooms + waterfront:bedrooms +
##     waterfront:centralAir + waterfront:bathrooms + bedrooms:sewer +
##     waterfront:sewer + fuel:sewer + lotSize:sewer + rooms:sewer +
##     livingArea:bedrooms, data = saratoga_train)
```

The handmade model was then compared to the current baseline model. Overall, the handmade model produced better RMSE results both relatively and absolutely, which can be seen below.

Table 1: Average RMSE of Each Model

	x
Baseline	66402.3
Handmade	66200.4

Furthermore, the boxplot allows us to determine that the handmade model generated a lower RMSE and a lower variation of the error values. This idea is reinforced by viewing the respective error probability distribution functions of each model.

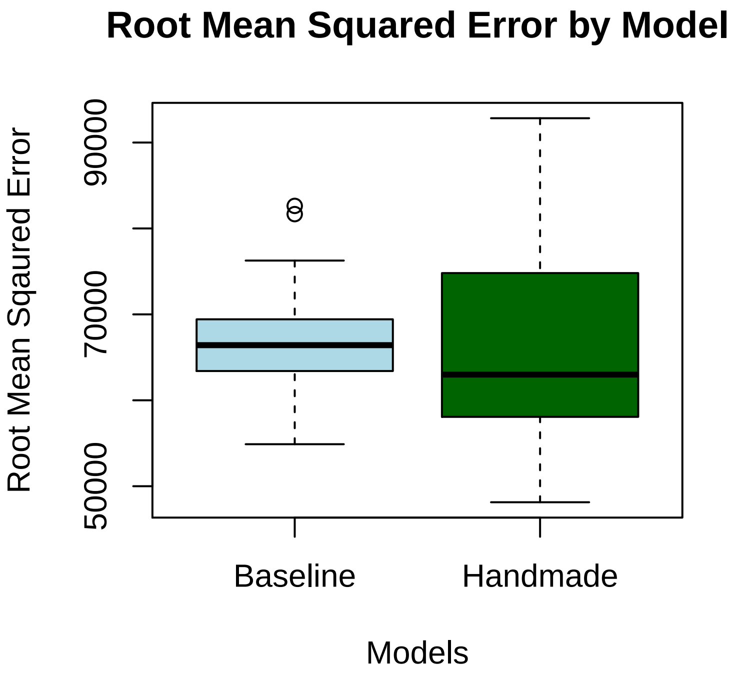
Table 2: Relative Benefit of Handmade Model v. Baseline Model

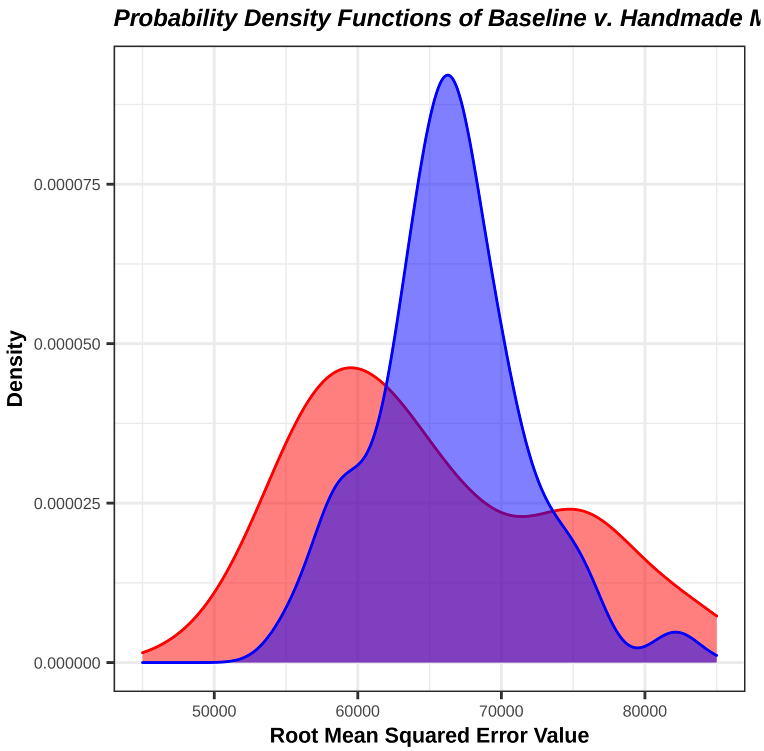
x
1.76%

Table 3: Absolute Benefit of Handmade Model v. Baseline Model

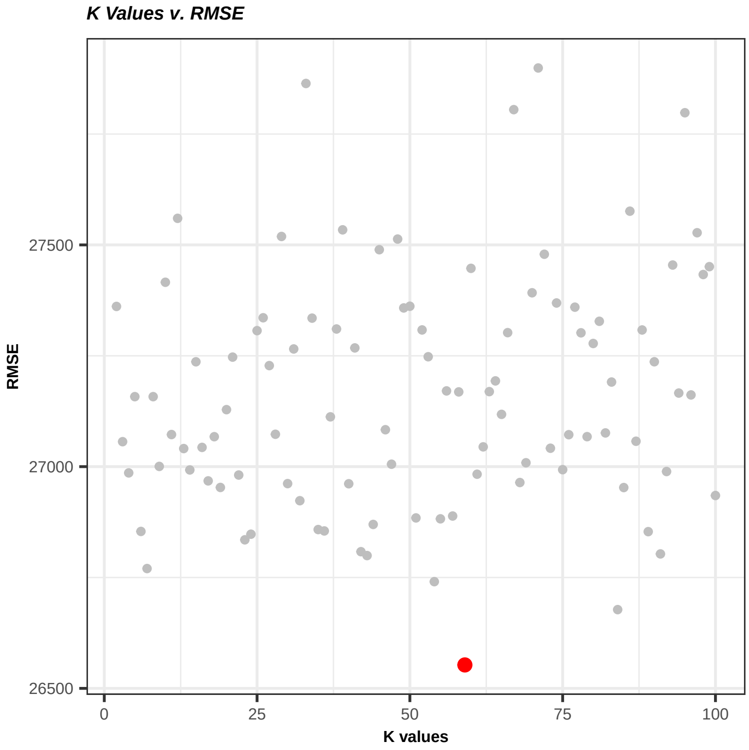
x
202

A current problem we are facing with our current model is the large variation of differences in the true v. predicted values of the house (especially in the tails) but the handmade model on average has a lower kurtosis than the baseline model, minimizing instances of extreme over/under valuation.





While the handmade model performed well, it was recommended we attempt to use the a K-nearest-neighbors model with the handmade models feature.



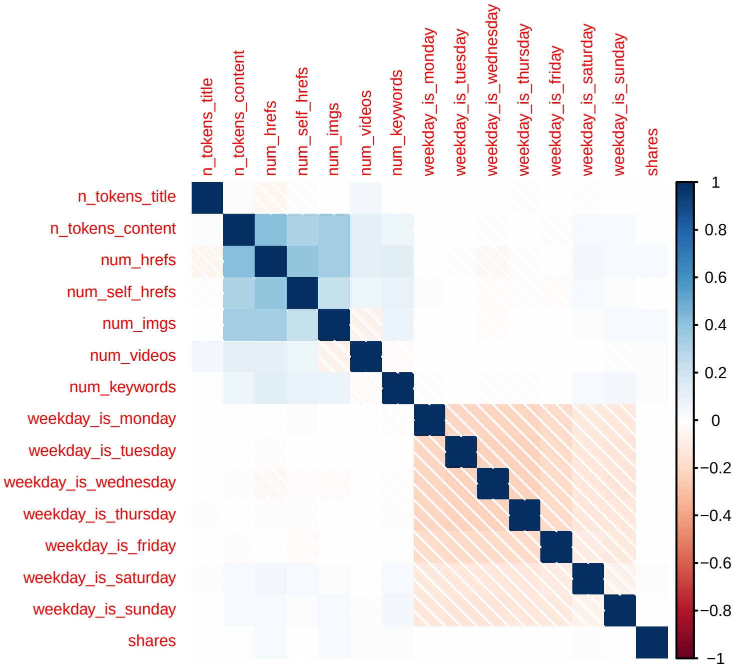
The KNN model performed even better than the handmade model. The best KNN model was determined with k=58 and produced an RMSE value of under 27,000. The implication of such a great model is our abilities to tax an appropriate amount based on a fair estimate of the underlying house. While there may be concern that the model performance was based on chance, it should be noted that the model was tested with 100 random simulations and thus we can assume the model does in fact produce reliable results.

(3) Viral Articles by Mashable

Mashable is a digital media and entertainment company whose main revenue driver comes from ads. As expected, a viral article posted will generate greater ad revenue as opposed to non-viral posts. Therefore, Mashable is interested in the ability to improve an article's probability of reaching viral status.

Mashable has provided us with online articles published between 2013 and 2014 with a list of features of each article. The question posed to us is: What features best determine viral status on an article and does this improve chances of a new article becoming viral?

To best determine how much an article will be shared, we must determine which features best are correlated with shares (both positive and negative). Below is a correlation matrix of the features of interest.



We created two model: a linear handmade model and an interaction-forward selection model that changes dynamically based on the training data. Both models use linear regression to predict the amount of shares of an article and then classified into viral or not based on more than 1400 shares. Below are the result of the models (both in and out of sample).

Now these model are compared to a null model - a model that predicts 'viral' regardless of features. Below are the results for null model.

The out-of-sample accuracy of the null model actually predicts viral status better than the handmade and forward selection models. Therefore, both models should not be used by Mashable as a predictor to determine viral status.

Table 4: In Sample v. Out of Sample Accuracy

	Average In-Sample Accuracy	Average Out-Sample Accuracy
Handmade Model	0.533	0.536
Forward Selection Model	0.533	0.536

Table 5: Model Performance Metrics In Sample and Out of Sample

	Average Error Rate	Average True Positive Rate	False Positive Rate	False Discovery Rate
Handmade Model - In Sample	0.467	0.999	0.999	0.467
Handmade - Out of Sample	0.464	0.999	1.000	0.463
Forward Selection Model - In Sample	0.467	0.994	0.993	0.467
Forward Selection Model - Out of Sample	0.464	0.995	0.996	0.463

Table 6: Null Model Performance

	Out of Sample Accuracy
Null Model	0.537

Now, rather then creating a model then adding a threshold to determine viral based on more of 1400 shares, we will classify viral (binary 0 or 1) then regress on certain features.

Again, we will generate two models: a handmade model and a forward selection model. However, both will be regressed on a binary outcome of viral status. Below are the results for both the models.

Table 7: In Sample v. Out of Sample Accuracy

	Average In-Sample Accuracy	Average Out-Sample Accuracy
Handmade Model	0.550	0.554
Forward Selection Model	0.579	0.586

Table 8: Model Performance Metrics In Sample and Out of Sample

	Average Error Rate	Average True Positive Rate	False Positive Rate	False Discovery Rate
Handmade Model - In Sample	0.450	0.443	0.347	0.447
Handmade - Out of Sample	0.446	0.453	0.346	0.436
Forward Selection Model - In Sample	0.421	0.388	0.235	0.384
Forward Selection Model - Out of Sample	0.414	0.406	0.236	0.370

As seen above, the classification model does significantly better than the handmade model, linear forward selection model and the null model by producing better results with in and out of sample accuracy, true positive rates, false positive rates, and false discovery rates. Therefore, we can conclude the classification model approach is better. The logistic classification model performs better due to the nature of the outcome. Since our purpose was to determine viral status of an article (binary outcome), using a linear regression (with the assumption of continuous data) would not be sufficient in determining a yes/no response. However, if we run a binomial logistic regression model, we are rather forcing a placement within the yes and no bucket.