# Exercise 2

Richard Marks

3/11/2020

Question 1: KNN Practice

Our goal for this project is to build two predictive models for the price of a Mercedes S Class car given its mileage. The two different models are based on two of the different types of trim 350 or 65AMG.

Lets begin by finding which value of K in our KNN regression will yeild the lowest Root Mean Squared Error (RMSE) for the 350 Trim. The K with the lowest RMSE has the least amount of error and can most accurately predict prices based on mileage. Lets start by running a few regressions with different values of K, and getting their subsequent Root Mean Squared Error (RMSE) which will be displayed for every value of K.

```
## [1] 10980.89
```

Starting with K=3

```
## [1] 9847.728
```

K=5

```
## [1] 9683.586
```

Jumping to K=9

```
## [1] 9594.833
```

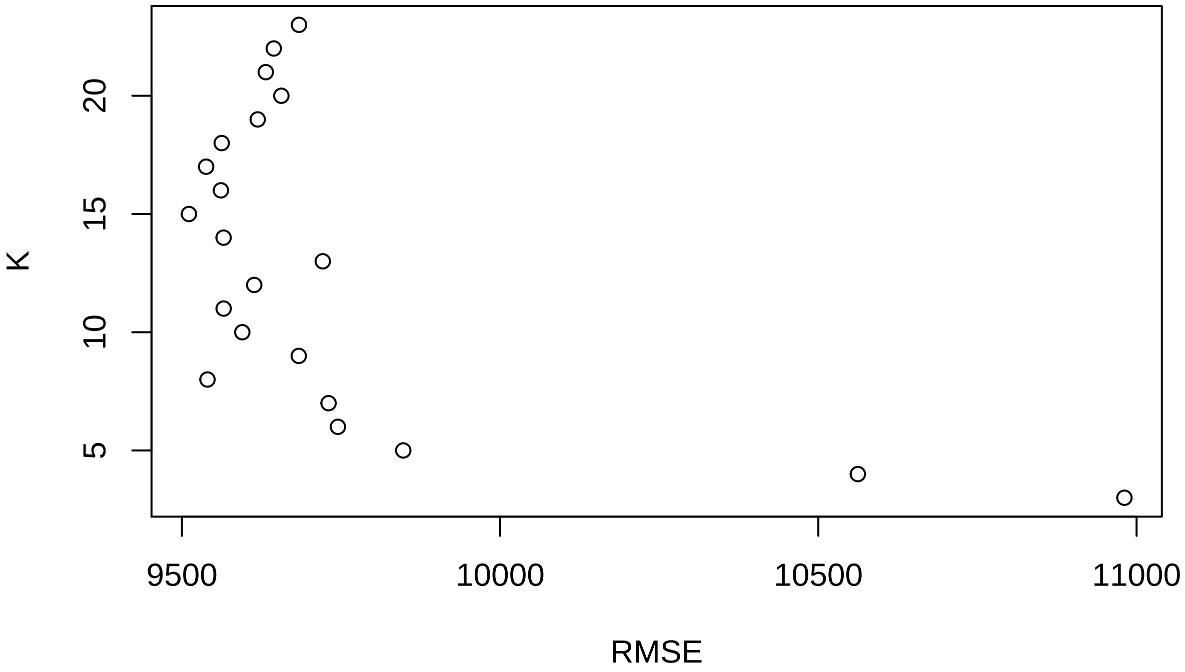
k = 10

```
## [1] 9565.566
```

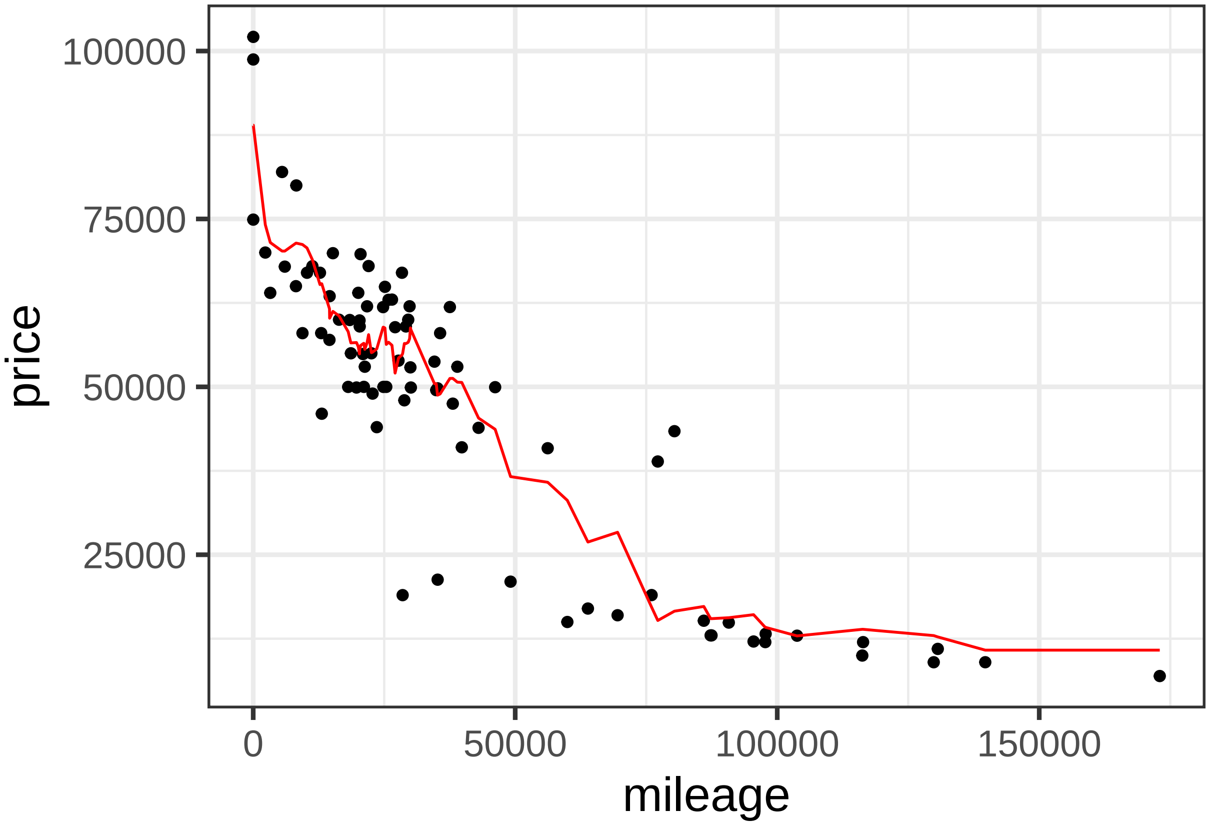
k= 11

```
## [1] 9511.004
```

and finally K = 15. Calculating all of these individually is inefficent so instead we should create a model featuring every single value of K from 3 to 23 and its appropriate RMSE.

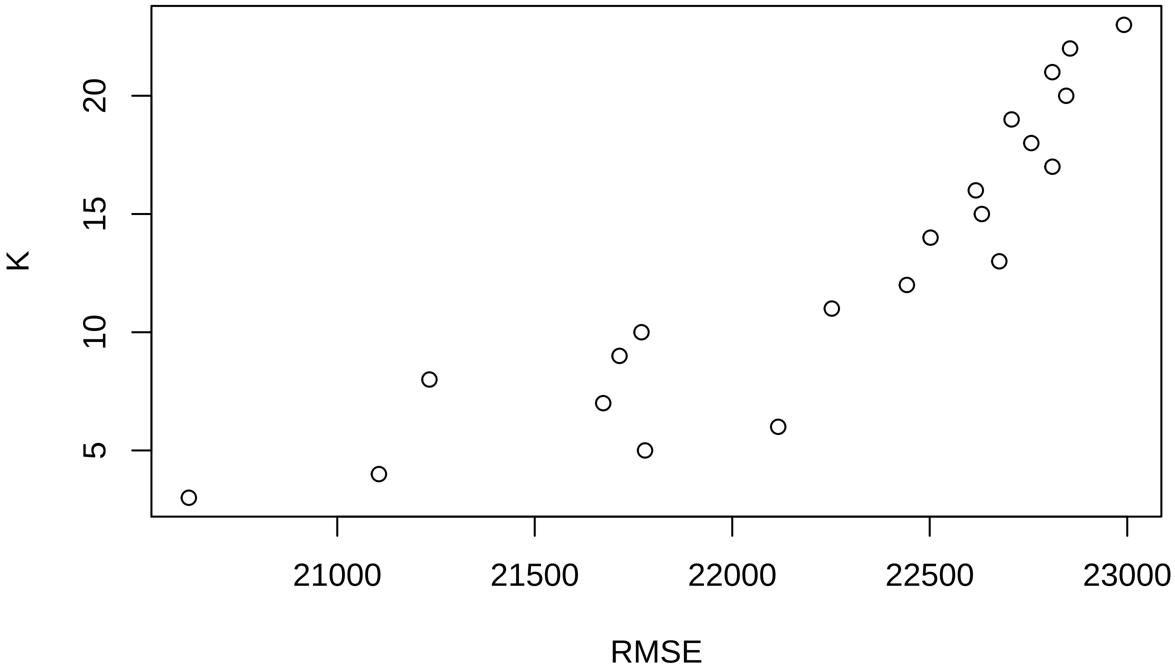


The plot above has K values 3-23, and their average RMSE after having been run 250 times each by Rstudios foreach do function. As you can see above, for the 350 category of Trim the optimal number of K nearest neighbors for a regression is 15 Now that we know this, we just need to plot the fitted model for K=15 which is shown below.

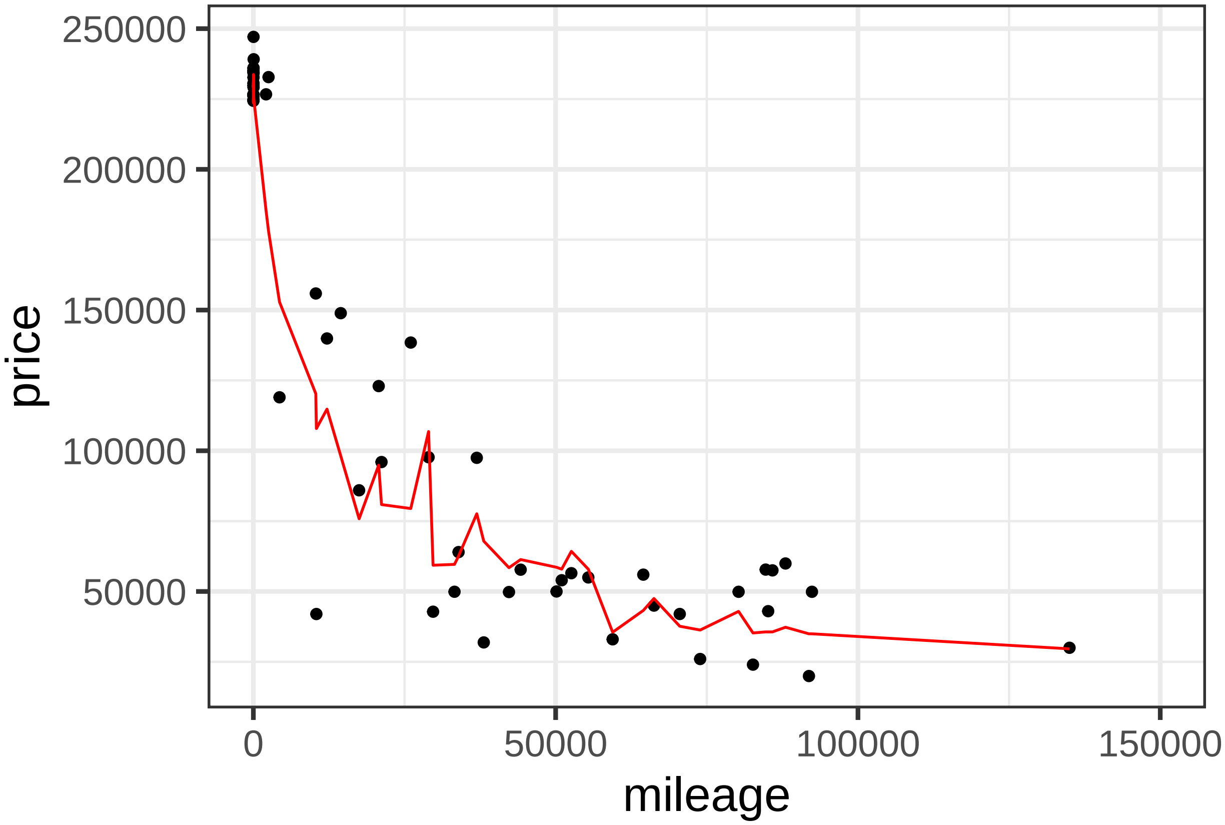


Now that we have a predictive model for the 350 Trim we need to repeat our actions and find the optimal K regression value for the 65 AMG Trim.

Since for the 350 trim we showed that calculating individual K values is inefficient we will go ahead and skip that step, instead plugging the 65 AMG trim data into the same equation used to create the K vs RMSE graph for the 350 Trim group shown below.



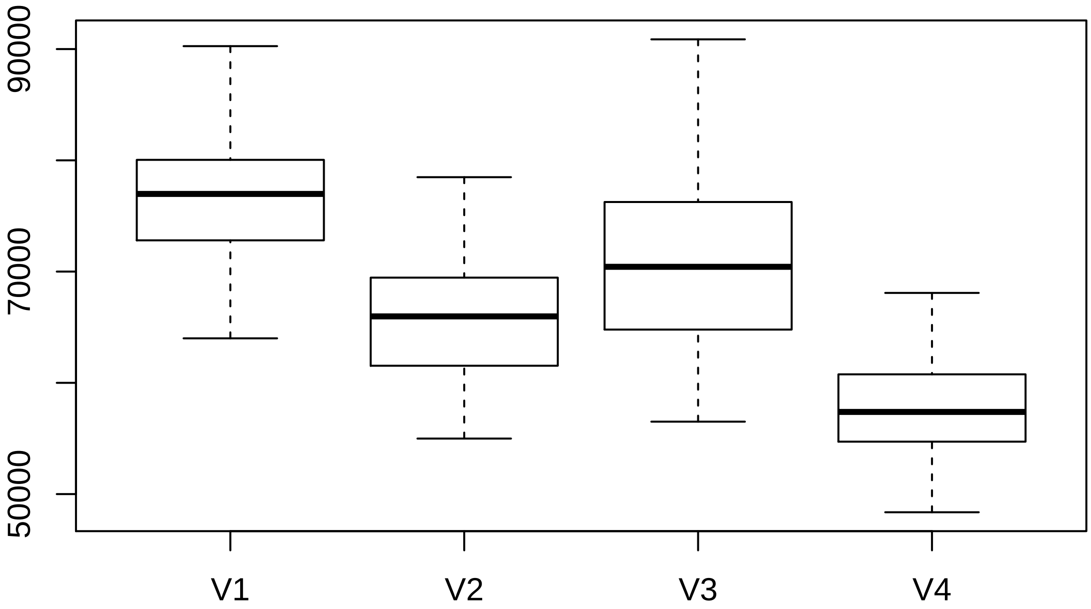
by looking at this graph we can see that the optimal amount of nearest neighbors for the 65 AMG group is 3. The lowest amount calculated. Now that we know the optimal value of K we can plot a fitted regression model for price based on mileage with a K=3 nearest neighbors regression, shown below.



The trim that had the highest optimal K value was the 350 trim class. The reason that the 350 trim had a higher optimal K is that there were more 350 trims than there were 65 AMG trims. The idea of a KNN regression is to look at the K nearest neighbors to each point to try to figure out the closest values. When we have more observations we need to look at more neighbors (K's) because there is a higher chance for error. If you use a larger, but not too large, number for larger data sets you decrease the chance of error. This explains why the 350 Trim class has a higher optimal K value than the 65 AMG class.

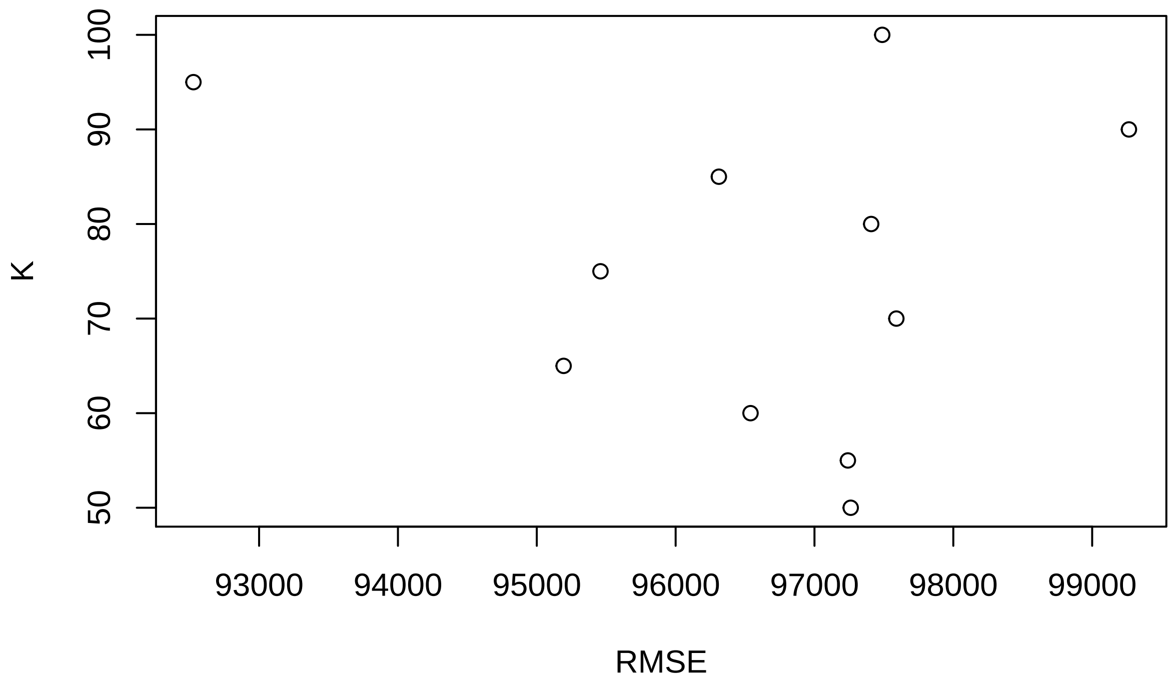
Question 2: Saratoga House Prices

What properties of a house effect its market value? Is it how many rooms? How many square feet? Whether or not it has a fireplace? To be able to answer this question we first need to look at the data. For this we will be using widely acessible data in houses in Saratoga. To try and find predictors of price I fit a series of linear regression models to the data. The first (V1) was an uncomplex model, only looking at a houses lot size, the amount of bedrooms, and the amount of bathrooms. The Second (V2) was slightly more complicated, looking all columns of the data set excluding, the type of sewer system, if it is waterfront property, the landvalue, and wheter the house has new construction. The third (V3) was the same as V2, except it also took into account the interactions between all of the variables, such as the interaction between the amount of bedrooms and bathrooms. The final model (V4) is very much based off of V2, except I used a step function to find the most ideal combination of predictors and the most ideal interactions. To see which model most accurately predicts house prices I took the Root Mean Square Error (RMSE) of all four models, in which a lower score means there is less error in the model.



The boxplot above shows the average RMSE of each model. As we can see V4 (RMSE=56,949.67) is significantly lower than V1 (RMSE= 78,078.15), V2 (RMSE= 67,404,01), and V3 (RMSE= 77572.35). This shows that of our four models V4 is the most accurate in predicting the price of a house.

Now that we have a pretty good predictive model the question comes to, is there a better possible model? So far I have only created linear regression models, which are not the only type of model available. Another available model is the K Nearest Neighbors regression model. For this model I ran all of the non categorical data excluding the house price in a standardized KNN from K=50 to K=100, going by every 5 neighbors.



The plot above shows that the optimal amount of K Nearest Neighbors is about 55, as it does fluctuate depending on how the training and testing split turns out. However as we can see the even the K value with the lowest RMSE has significantly higher amount of error than our V4 model, showing that our V4 linear regression is the optimal model to predict the price of houses.

From V4 there are three variables that are the largest indicators of a houses price: The Value of the land the house is built on, the amount of living space in the house, and oddly enough the amount of fireplaces that are in the house. These factors had the most amount of interactions with other factors in the model. If you use this model on other sets of houses you can get a good prediction of a houses market value.

Question 3: Predicting When Articles Go Viral

In this age where the internet is an integral part of the lives of the majority of the worlds population, there is an important question for many internet news sites, how does an article go viral? It seems that nearly anything could make an article go viral, the title, the pictures, maybe just dumb luck. But statisitcally we can find a formula that at least can show when an article can go viral more than a 50/50 chance. After creating a linear regression using many different aspects of a post, and the interactions between them (such as the interaction between how many pictures are in the article and how many videos are in it) we need to create two categories for the outcomes. If the post has 1400 shares or more then the post is considered viral, and if it has less then it is not viral. Now that we have this binary we can create a confusion matrix to see how well our model does.

```
##             Predict_Viral
## Actual_Viral    0    1
##            0  498 3590
##            1  208 3633
```

Looking at the above confusion matrix we can see that our error rate is 47.5%, 2.5% better than if we had just predicted that no aritcles went viral. Our model is very good at predicting if an article will go viral (true postivie rate= 94.8%), however where our model fails is that it predicts most articles to be viral even if they are not, (false positive rate= 88.9%).

Now that we have run a linear regression , it is important to see if there is maybe a different modeling strategy that may have greater results. For this we will run a classification K nearest neighbors regression. To do this we had to create an entirely new variable, a binary variable Viral, where it can either be viral or not be viral. After running multiple KNN, we found that the optimal number for KNN is 150. Below is the confusion matrix for the KNN regression.

```
##              PredictedViral
## ObservedViral     0     1
##             0 13489  5925
##             1  9140  9708
```

As you can see from our confusion matrix above, our overall error rate was 39.3%, 8.2% lower than our previous model. Our false poitive rate is significantly lower than our previous rate (30.6%) , but with that our true positie rate also lower significantly (52%). This classification model is the superior model, because its overall accuracy is higher. The first model was way too willing to predict an article to go viral, which is not accurate to reality and could lead to articles with lower shares being promoted because they were predicted to go viral. The second classification model was very good at telling that an article will not go viral, showing restraint in classifying viral articles. The classification way was superior because it treated the issue like a classification problem the entire time. While the linear regression model only treated the end of the problem like a classification problem. This issue revolves around a classification binary, viral or not, and by treating the problem how it should be gave a better model.