You should also include a write-up for each data set. The write-up should describe (in a paragraph or so) what approach you used in coming up with your predictions. The discussion should be such that I could reproduce your analysis and thus re-calculate your predictions. Unless absolutely necessary, do not include R code in your write-up. (In markdown, use echo=FALSE)

You should also include at least one paragraph in your write-up describing your method for estimating the sum of squared residuals (SSE).

Dataset 1:

I will name all features a, b, c, etc. and response is just response.

I first went ahead and plotted all the features against the response variable to find any obvious hints. Instantly, we see that c has the largest correlation with our response variable, but transformations need to be done to make it a linear fit. Looking at the plot of c against our response, I can tell I need to run the exponential function transformation. I now found the correlation between a, b, c, exp(c), and the response variable. This helped to affirm my theory that c has high correlation with the response and the other variables are barely correlated. After running linear regression with the formula: , I now need to check if my assumptions hold. After plotting a histogram of the residuals, they seemed mostly normal if a little skewed to the right. However, plotting the residuals against exp(c) showed that our assumption of constant variance is not met.

I now need to transform our response variable. To do this, I wanted to run boxcox but found that the response variable had negative values; I then shifted the response over by the absolute value of the minimum response value and added a very small epsilon to have only positive values. I now ran boxcox on these shifted response values and it returned that the square root transformation would have the highest log-likelihood. I now trained the model with the formula: . I then checked for normality and constant variance on the residuals and they seemed much better after the Y transformation.

I then checked the SSE and MSE of the model by running an anova test and found them to be appropriately low. Additionally, I plotted the model against the transformed variables and it seemed like a very good fit on the training data. Now to check on the testing data, I had to call predict like normal on our test data but then squared the output and then subtracted the shift in order to get the normalized values. I then calculated SSE and MSE of the predictions by subtracting my predictions from the test response values, squaring those values, summing the values up, and then dividing by the number of observations. The MSE of the model (from the training data) and the MSE found from our predictions are very close which generally means the model generalized from our training data to our testing data fairly well.

Dataset2:

I will name all features a, b, c, etc. and response is just response.

First thing I did was to plot all features against the response variable. This proved not very helpful since all the plots just looked like noise with no real correlation between any feature and the response variable. Even running correlation between all features and response basically told me the same thing. I then used the professor’s hint to use the index and then plotted the index of the observation against the response and found a mostly linear plot with a weird cutoff at around index 100. This made me start plotting index against my other features to check if there was any correlation with that specific index. I immediately saw that if feature a was equal to 1, the response followed a whole different model. This prompted me to check the values of feature a in the test dataset and found feature a was always equal to 1. This meant that I could throw away all the observations where feature a was not equal to 1. This happened to be the first 100 observations. Additionally, I threw away feature a since this is now always 1. I then plotted the rest of the features (including the index) against the response values on the subset of the training data. These still looked like noise, but I could still see a slight correlation against the index. I calculated correlation once more and found that my hunch was right, and that index was still highly correlated. Running a F-test to check if its coefficient was 0 or not came out statistically significant in which we rejected the null hypothesis that index’s coefficient is 0. Thus, I now fitted a model using the formula . I then checked to see if the model’s residuals were normal and if the assumption of constant variance held. They did and so I moved onto checking the model’s SSE/MSE against the predictions’ SSE/MSE.

Calculating SSE/MSE with this model is much simpler than the previous model since no transformations were needed. We plug the features’ observations in from the test data, subtract the prediction from the real value of y, square it, sum them all up, and then divide by the number of observations to find our MSE. The MSE of the model (from the training data) and the MSE found from our predictions are very close which generally means the model generalized from our training data to our testing data fairly well.