schuler_5779_ece_527_report_03

September 26, 2024

1 GMU ECE 527 - Computer Exercise #4 - Report

Stewart Schuler - G01395779
20240926

1.1 Exercise 4.1

To aid in choosing which features to include in the linear regressor we can compute the correctlation of each feature with our *y* variable *mpg*. *Table 1* contains the correlation results, it can be seen that *weight* and *displacement* have the highest linear correlation levels with *mpg*.

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin
mpg	1.000000	-0.775396	-0.804203	-0.780255	-0.831741	0.420289	0.579267	0.563450

Table 1. Feature Correlation to mpg

Another was to visualize the correlation is to plot the correlation between all features. As seen in *Figure 1*, *cylinders*, *displacement*, *horsepower*, and *weight* all have a high correlation to *mpg* (and as a result with one another). While the remaining features have a much weaker correlation to *mpg*, and likewise have almost no correlation between themselves.

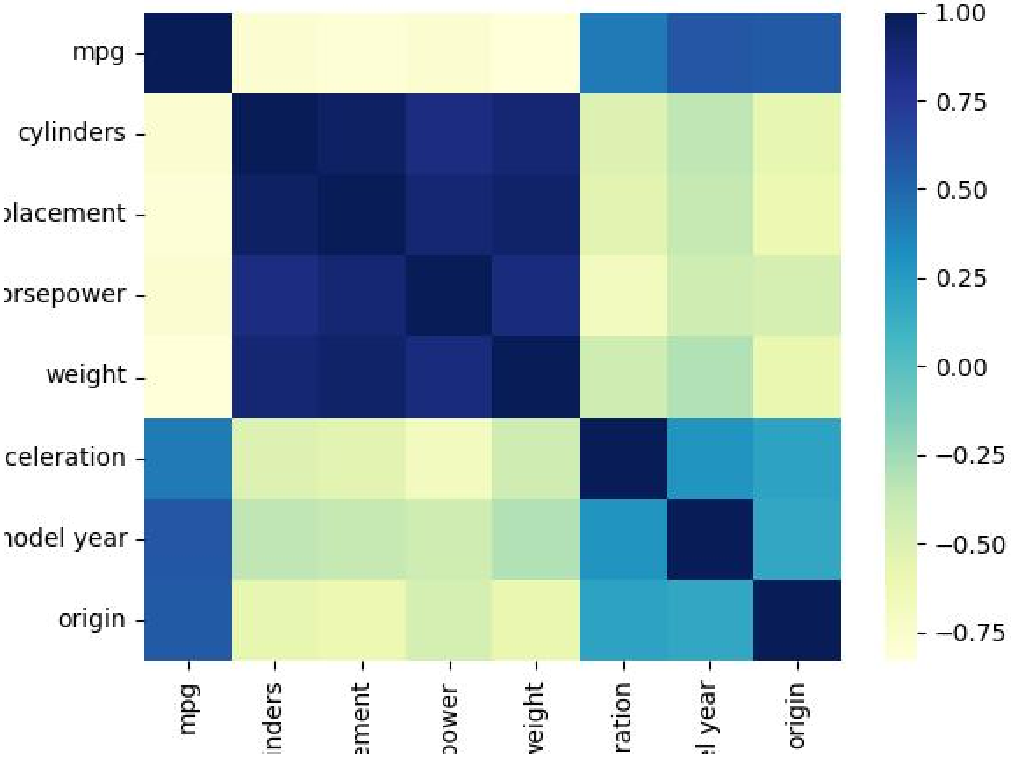

Figure 1. Correlation Heatmap

4.1.3 Questions **Question:** Based on what you have been able to learn about the data set, what features seem to be the best for predicting gas mileage?

Answer: The best way to predict quality of the linear regressor is the correlation values shown in *table 1.* From the table the best single feature would be *weight*, followed closely by *displacement.*

Question: Are there any features that seem to be irrelevant or not useful in predicting gas mileage? Whcioh ones are they and why would the not be useful?

Answer: *Origin*, that is manufacturer country has the lowest correlation to *mpg.* That can be seen when plotting it vs *mpg.* Of the three possible discrete values there is significant overlap. And while they do trend upward with enumerated *origin* there is still significant overlap. A case could be made that these could correlate with underlaying country emission laws, but that is a far weaker predicter than other dataset features. *Model year* likewise has a very slight correlation with *mpg* with a lot of overlap between feature values. Again this may not be entirly useless because we could reasonably make the assumption that as model year increases average *mpg* should also increase because that is desirable for manufacturers, but alone that *model year* feature doesn't take into account the type of vehicle, when used alone we could comparing a small commuter vehicle to a semi-truck and have now was to distinguish between without additional features.

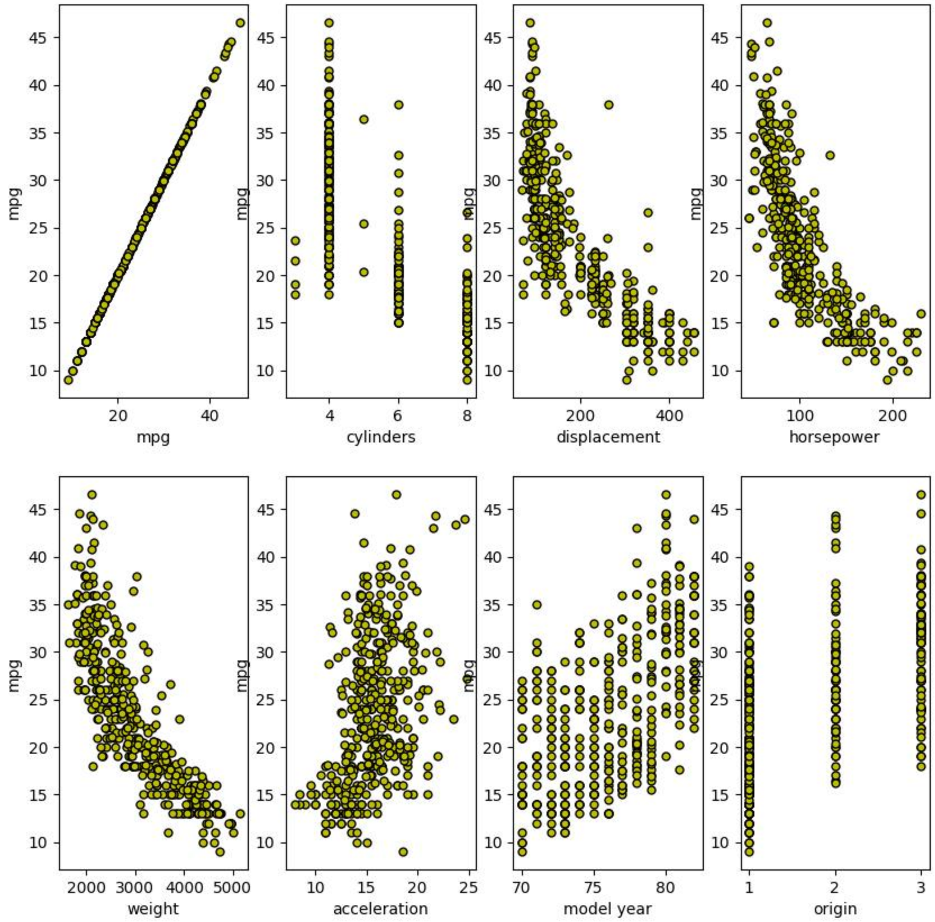

Figure 2. *mpg* vs *features*

1.2 4.1.4

Next we choose a specific single feature to run regression on to make a predictor for *mpg*. As dicussed in the previous section *weight* was chosen as the single feature.

4.1.4 Questions Question: Is it important or necessary to scale the data before performing regression?

Answer: Unlike the SGD algorithms previous studied the linear regression algorithm being used here does not require feature scaling. The is because we are finding the closed form solution to the problem. With the SGD algorithms we were estimating a value then updating by some step size based on the previous result. That step size update is highly dependent on data scaling. Since there is no step size for linear regression the algorithm performs the same independent of scaling.

4.1.4 Exercise Using *weight* as the only feature we produced the regressor shown in *Figure 3* with residual plot shown in *Figure 4*. As can be clearly seen from the residuals plot, as the *mpg* value increases the model becomes less accurate, by under predicting the value. From this it can be concluded that a linear model doesn't fit this dataset well, given that we expect *weight* to be the best performing linearly modeled feature. From the residual it can be seen that a predictor with a polynomial like curve would reduce the residual error to the desired gausian distribution around 0.

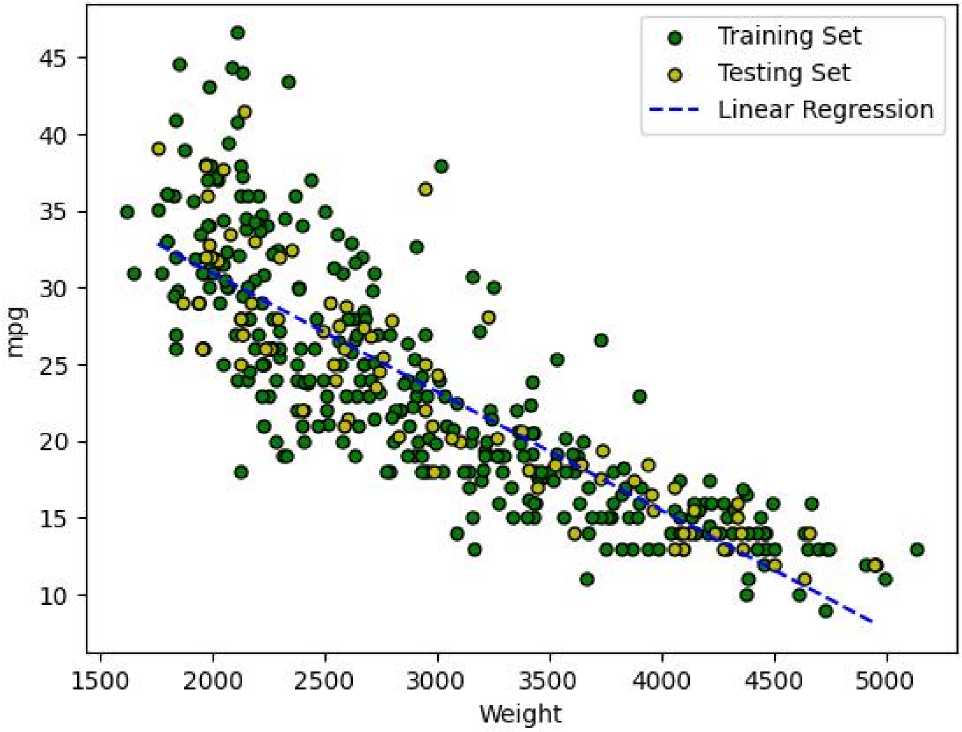

Figure 3. *mpg* vs *weight*

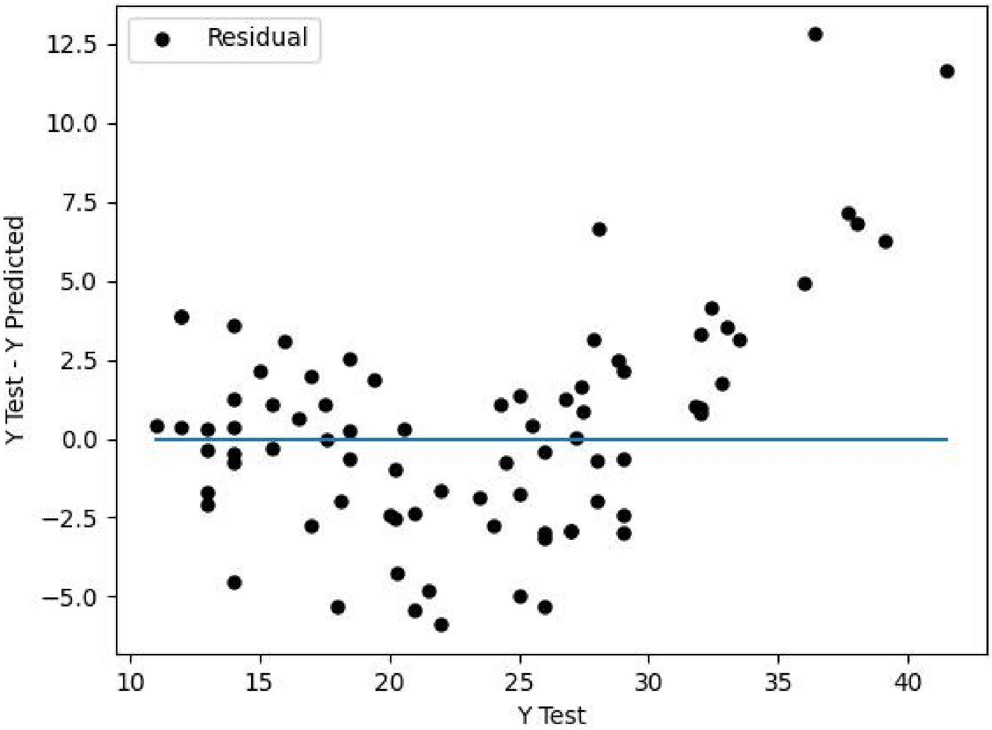

Figure 4. *weight* Regressor Residual

The experiment was linear regression was repeated using the expected second best performing feature *displacement*. This too sufferse from under predicting as *mpg* increases. However the similarity between the results of the two features confirms what the correlation values predicted. That since the two feature sets are highly correlated their results should be as well. The R^2 score and mse for both predictors can be found in Table 2, included in that table is the results of a regressor using the *origin* feature. It's inclusion demonstrates that the linear correlation table is a good predictor of performance for the linear regression model.

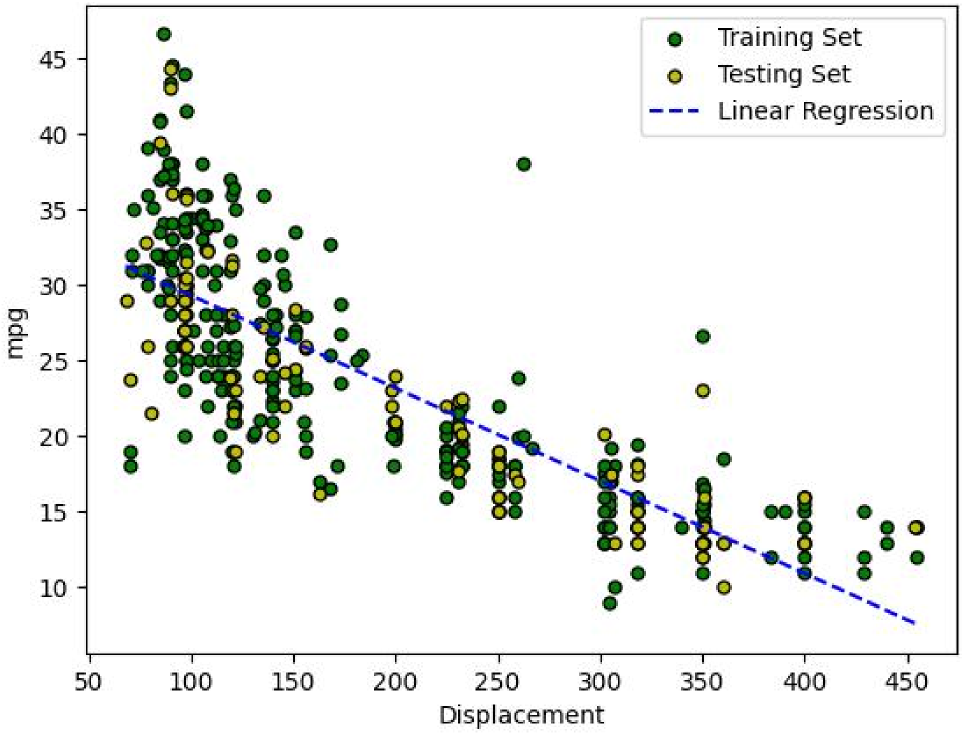

Figure 5. *mpg* vs *displacement*

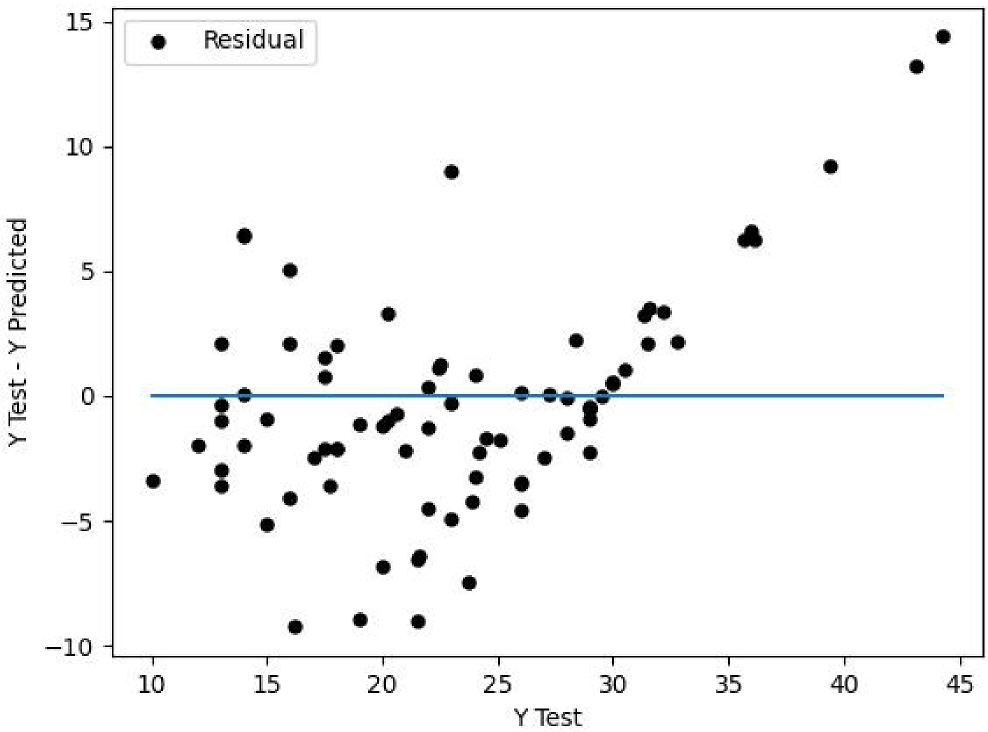

Figure 6. *displacement* Regressor Residual

feature	R^2 Score	MSE
weight	0.671	18.503
displacement	0.637	17.532
both	0.726	15.511
all	0.772	10.944
origin	0.308	34.617

Table 2. Regressor Performance

We consider next multi-variable regression. For the two variable regressor we combined weight and displacement. *Figure 7* shows the regressor model in each feature space. The two variable regressor produced scores shown in the *both* row of *Table 2*. As expected it out performs both of the individual parts, since each feature represents some information not captured by the other, but relevant to *mpg*.

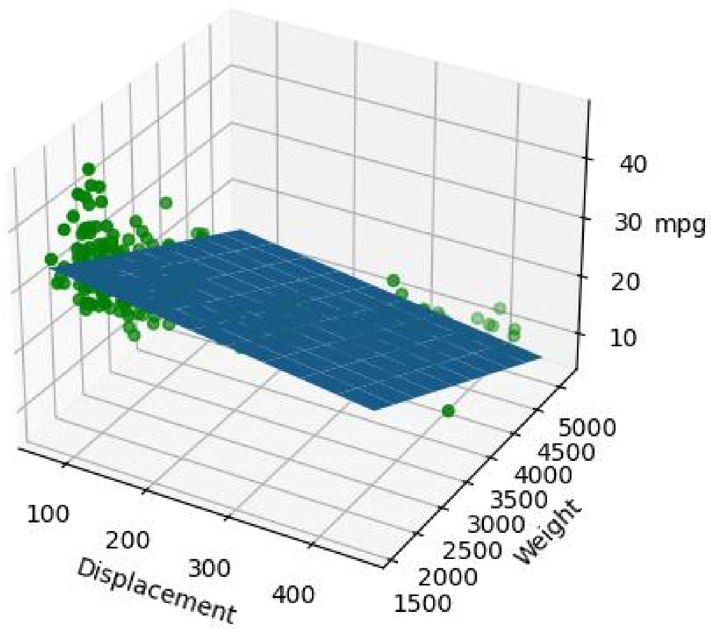

Figure 7. *mpg* vs *displacement* & *weight*

The final linear regressor tested on this data set considers *all* available features. The score results are in the *all* row of *Table 2*. This model performs the best of all the models tested so far. However it only marginally out performs the two feature model. This is because some of the additional included features have lesser correlation to *mpg* and likely don't contain useful information. Where as some of the newly include features like *horse power* and *cylinders* are decently correlated to *mpg* and will slightly improve the prediction.

However it should be noted that because the dataset is rather small, there is a good sized variance between experimentally measured score values depending on the random seed.

1.3 Exercise 4.2

2nd Order Polynomial KRR Consider next a non-linear regressor. In this experiment the Kernel Ridge regressor with a 2nd order polynomial kernel was applied to the *weight* feature. As was previously shown in *Figure 3* a linear model doesn't quite fit the bend in the dataset. *Figure 9* shows the regression plotted verses the datasets. Visually it appears that the regression model fits the data better than the linear model (*Figure 8*). This is further confirmed by comparing the R^2 and *mse* scores in *Table 3*.

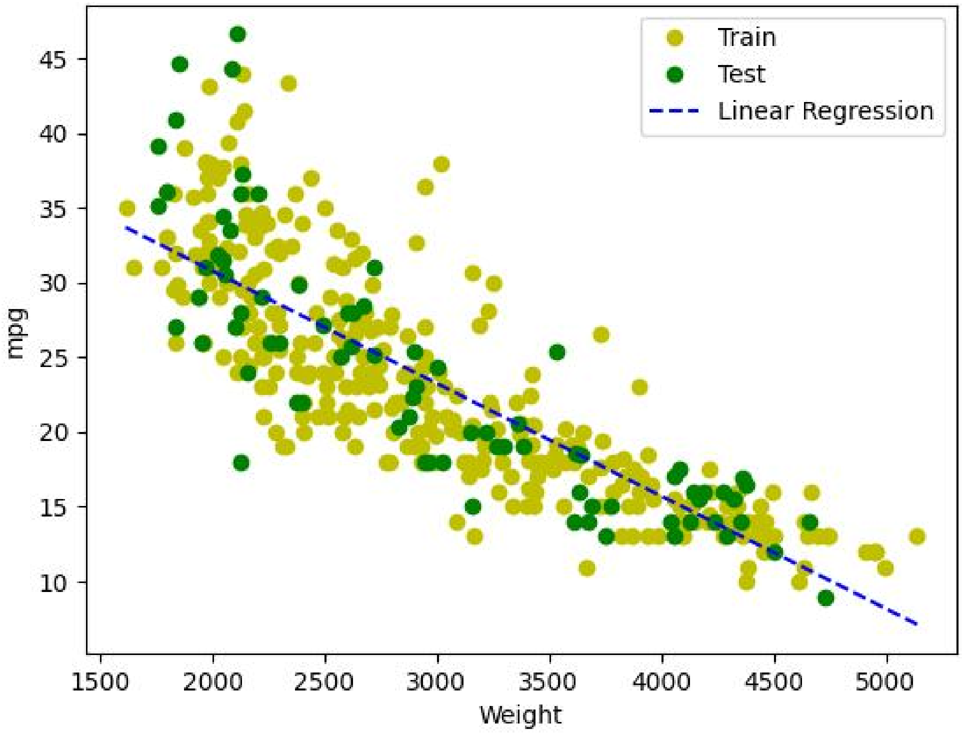

Figure 8. Linear *mpg* vs *weight*

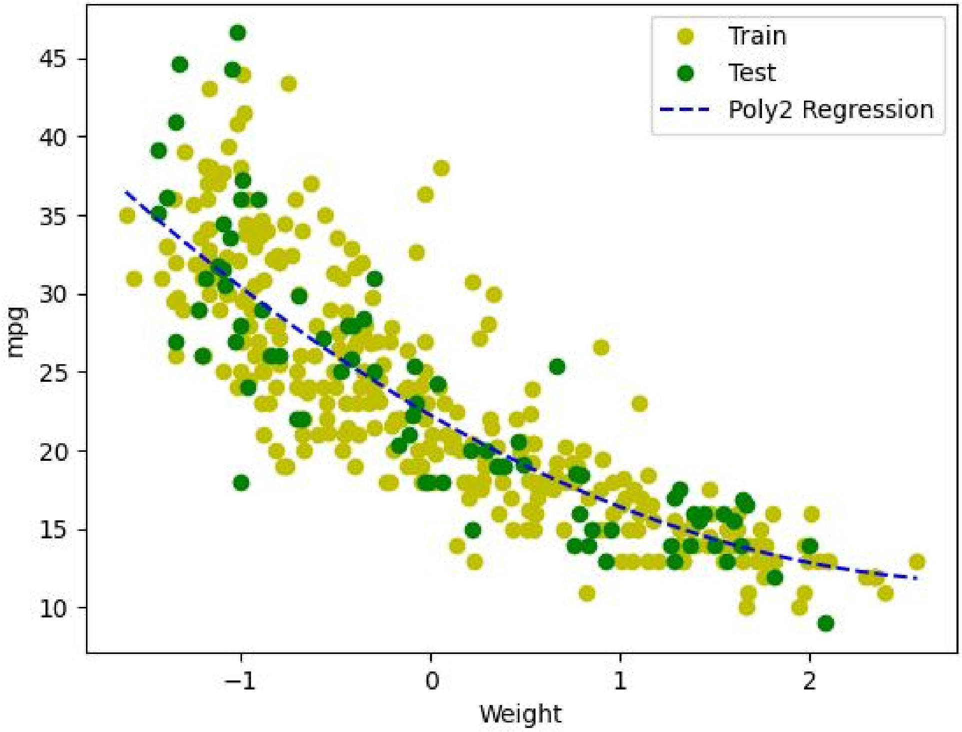

Figure 9. 2nd order polynomial *mpg* vs *weight*

model	R^2 Score	MSE
linear	0.646	19.149
poly 2	0.669	17.886
poly 3	0.669	17.912

Table 3. KRR Results

Next consider the impact of the α parameter on the regressor. *Figure 10* compares the R^2 score for the 2nd order polynomial regressor for different values of α based on the same train/test split. As can be seen as α increased the score decreases. I is observed that there is a sizeable and consistent offset between the training and testing curves this indicates that the regressor may be overfitting to the training dataset. However in this specific case, since the model is only a 2nd order polynomial which can be hard to overfit to this type of dataset, the offset is more likely a result of the **specific** train/test split used for this experiment.

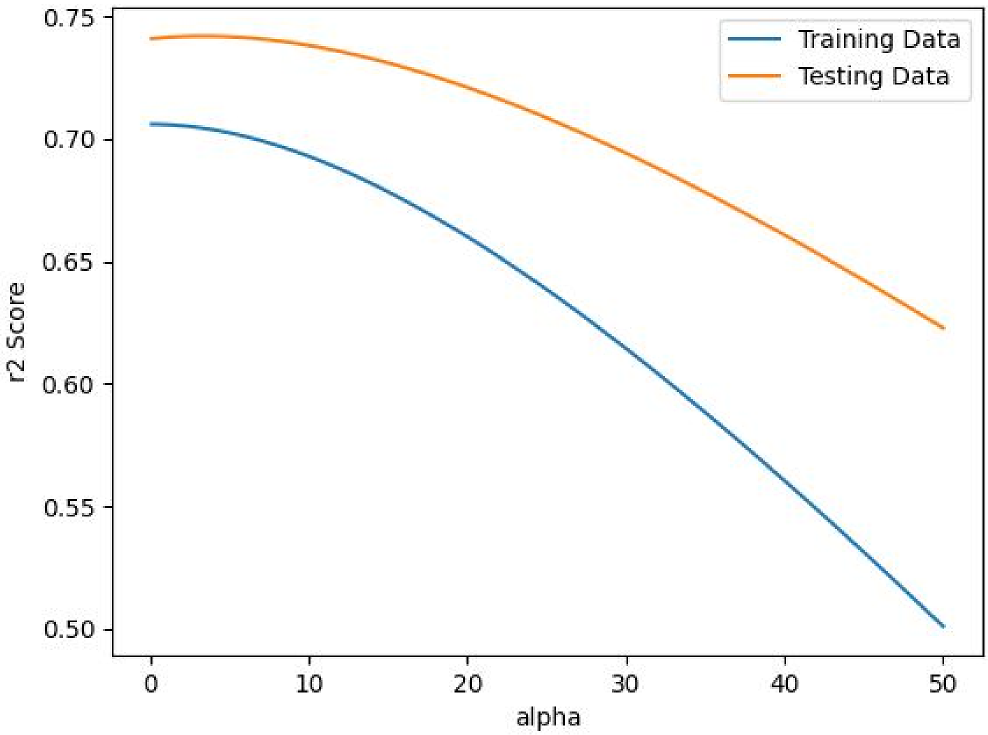

Figure 10. 2nd order polynomial R^2 vs α

3rd Order Polynomial KRR Lastly consider now and 3rd order polynomial kernel for the KRR regressor. Applying the regressor to the *weight* feature produces the model shown in *Figure 11*, unsuprisingly given the shape of the dataset the model produces a polynomial very similar to that of the 2nd order regressor. Likewise the α curve is also very similar, since for a model such as the one shown in *Figure 11* the 3rd order coeffient is very small compared to the other two.

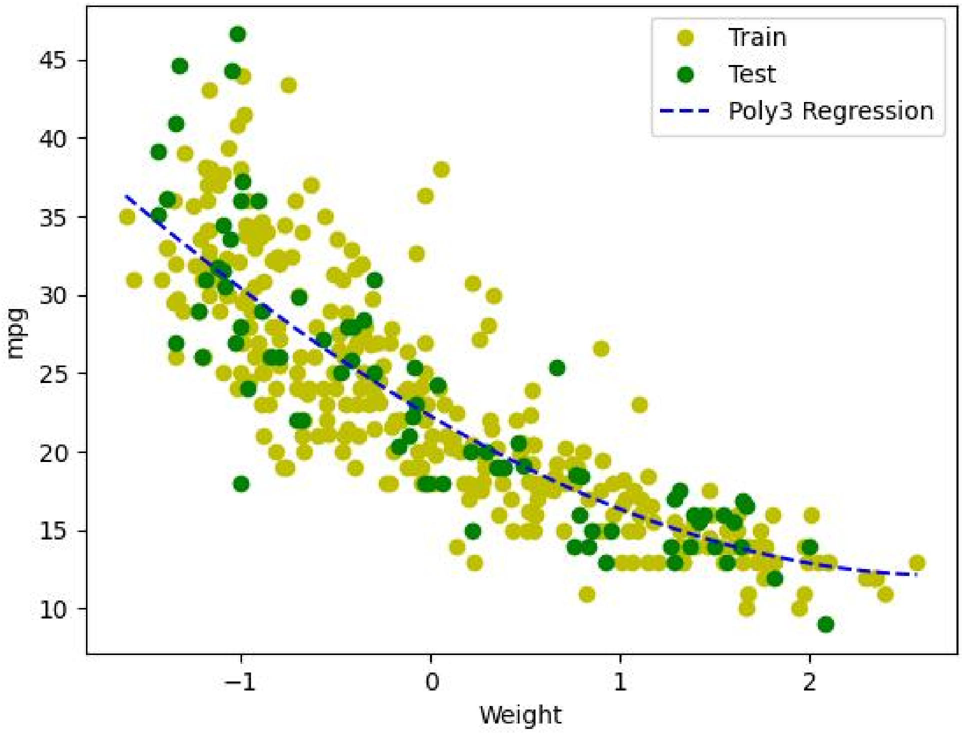

Figure 11. 3nd order polynomial *mpg* vs *weight*

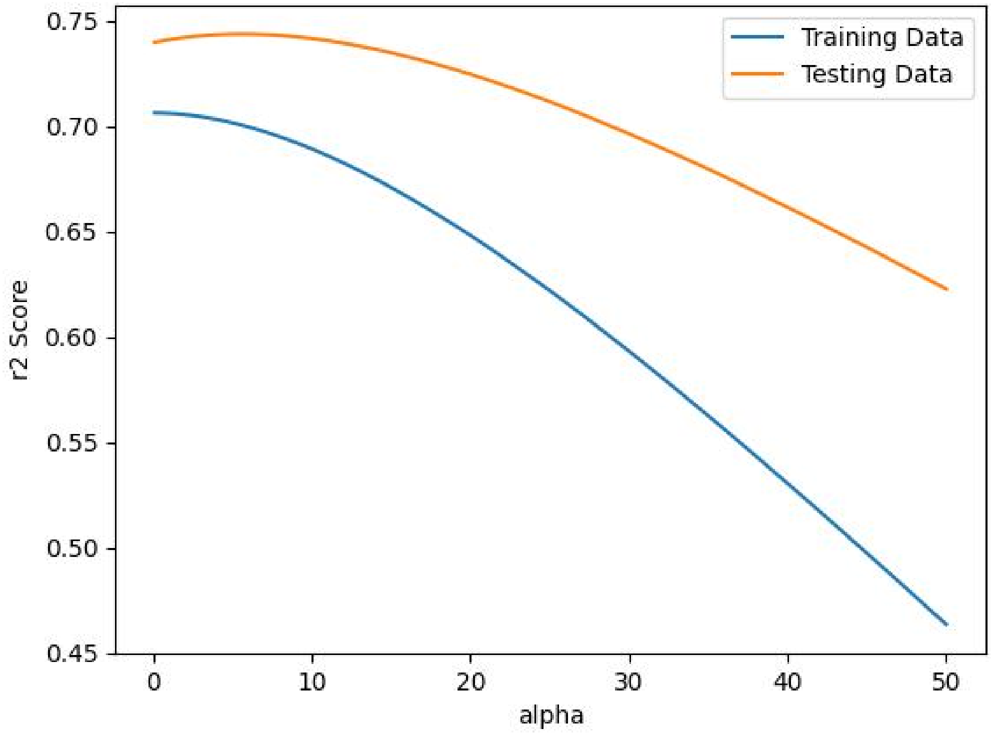

Figure 12. 3nd order polynomial R^2 vs α

Finally we consider the the residual plots for these two non-linear models. They are nearly identical and share the same trends as the linear residual plot previously discussed. Given that the data has a polynomial like curve to it, I would've expected the polynomial kernels to perform better than they did. I believe the results reflect what they did because of the smallness of the dataset, with more data the scores will be less sensitive to the randomness of the train/test split.

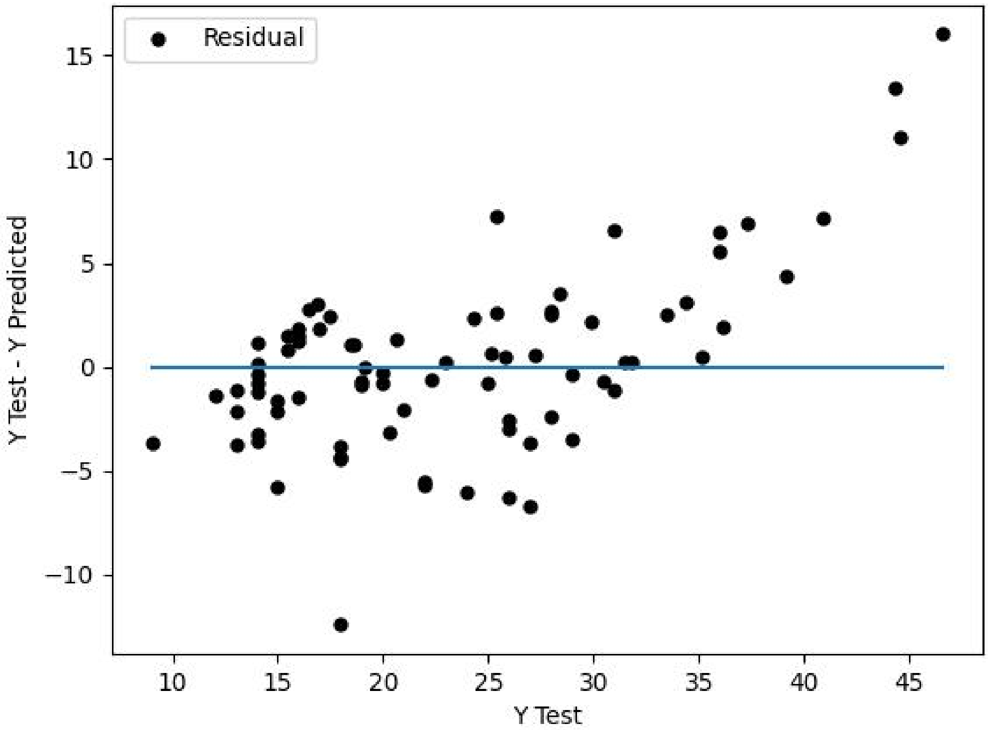

Figure 13. 2nd order polynomial residual

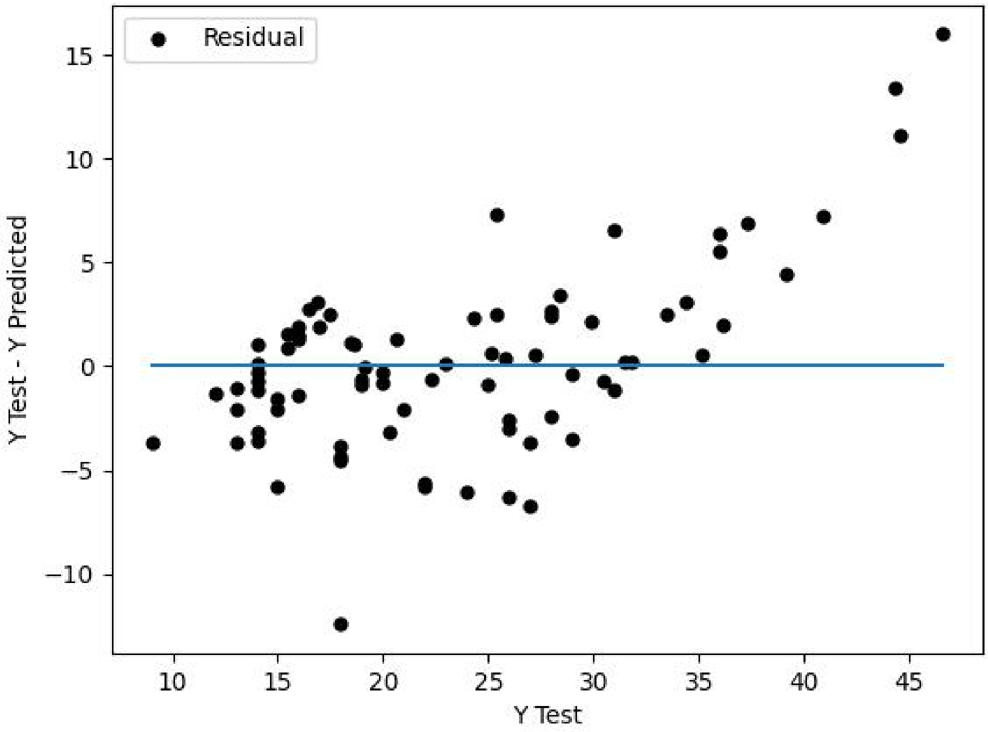

Figure 14. 3nd order polynomial residual