COSC 528 Project 1: Multivariate Regression for Car Gas Mileage Estimation

Elliot Greenlee

October 5, 2017

Abstract

This report covers the design and implementation of various multivariate regression methods in order to predict gas mileage. Linear and polynomial regression were used on eight features of automobile data including model year, weight, and horsepower. In order to improve accuracy and provide insight into the data, statistical analysis, normalization, and dimensionality reduction were performed. The highest accuracy achieved on this task was an R^2 value of 0.86 using basic quadratic regression, with car weight having the greatest effect on the gas mileage.

1 Introduction

Humans make immediate predictions about variables constantly throughout the day. These predictions are often accurate due to correlations in seemingly unrelated data. Computational power allows for more nuanced exploration of these correlations not easily understood by a human. In the case of this dataset, automobile information for predicting car gas mileage, features include weight, horsepower, and model year [cmu]. Regression methods are able to use these numeric values for prediction in a way human ability cannot. In regression, given an input x_i, the output r_i is a numerical value, and a numeric function $f(x_i)$ should be learned. In machine learning, this function is learned from a set of i training examples with assumed noise ϵ as $r_i = f(x_i) + \epsilon$, and estimated by $g(x_i) = w_0 + w_1 x_i$. The error on the samples is calculated as $E(g(X)) = \frac{1}{N}\sum_{i=1}^{N}[r_i - g(x_i)]^2$. By taking the partial derivatives with respect to the weighting variables, and setting them equal to zero, the minimum can be solved for.[Alp14]

The objective of this project is to implement multivariate linear and polynomial regression methods and apply them to automobile data in order to predict gas mileage. Additionally, optimization techniques such as normalization and dimensionality reduction are attempted in order to further improve the performance. The highest accuracy achieved was an R^2 value of 0.86 using basic quadratic regression.

2 Data

The data for this report is a set of automobile features concerning city fuel consumption. Three attributes (cylinders model year, and origin) are multivalued discrete and five (mpg, displacement, horsepower, weight, and acceleration) are continuous. There are 398 total instances, six of which have been removed because of a missing horsepower value. Each instance is a car with a unique textual label that is not used for analysis. This information comes from the StatLib library maintained at Carnegie Mellon University and was originally contributed by Ernesto Ramos and David Donoho [cmu].

2.1 Statistics

Feature	Min	Max	Mean (mode, count)	Std Dev
MPG	9.0	46.6	23.4	7.8
Cylinders	3	8	194.4 (4.0, 199)	1.7
Displacement	68.0	455.0	194.4	104.5
Horsepower	46.0	230.0	104.5	38.4
Weight	1613.0	5140.0	2977.6	848.3
Acceleration	8.0	24.8	15.5	2.8
Model Year	70	82	76.0 (73.0, 40)	3.7
Origin	1	3	1.6 (1.0, 245)	0.8

Table 1: Minimum, maximum, mean, and standard deviation for each feature. Mode and count is shown with mean for discrete features Cylinders, Model Year, and Origin

2.2 Preprocessing

Different variables can all have wildly different ranges and variances, meaning that weights for different variables are not consistent in meaning. In many machine learning methods, such as clustering, this can have a detrimental effect on the ability to learn a model. For this project, z-normalization is used to transform the inputs into outputs with a mean close to zero and a standard deviation close to one. This is done for each input x_i using $z_i = \frac{x_i - \mu_i}{\sigma_i}$, where μ_i is the mean and σ_i is the standard deviation of the data.[Alp14]

Although in theory a greater number of statistically independent features should reduce further and further the error, in practice additional features often lead to worse performance. This occurs either when the wrong model is used, or because the training sample is not infinite, so the distributions are not accurately estimated. This leads to problems with overfitting, where the model works well on the training data but poorly on the testing data, and complexity, where the model takes longer to run at $O(d^2n)$ where d is dimension and n is number of samples. The solution to these problems is dimensionality reduction. Linear methods are used because of their simplicity, projecting high dimensional data onto a lower dimensional space.

One common method is principal component analysis, with the objective of minimizing information loss in a least-squares sense. Principal component analysis attempts to reduce the data while minimizing information loss. Typically a maximum allowable loss is chosen. One might imagine the major and minor axis of a set of two dimensional elliptical data. Principal component analysis would choose the major axis as the projection direction to preserve information. Because there is no consideration of class information, it is an unsupervised method. The projection vector is obtained using the Karhunen-Loeve theorem, by discarding the minimum orthogonal vectors. This is possible by obtaining the eigenvectors and values of the covariance matrix, and then discarding the smallest eigenvalues and corresponding vectors. This minimizes the squared error.[DHS01]

3 Multivariate Regression

In multivariate linear regression, the numeric output r is produced by a weighted sum of several input variables $x_1, x_2, ..., x_d = \vec{x}$ and noise as

$$r = g(\vec{x})w_0 + w_1x_1 + w_2x_2 + ... + w_dx_d + \epsilon$$

Now the empirical error on the sample set is

$$E(g(X)) = \frac{1}{N}\sum_{i=1}^{N}[r_i - g(\vec{x}_i)]^2$$

but the same method of taking the partial derivatives with respect to the weights and setting them equal to zero is possible. This results in a set of normal equations which can be written as $X^TX\vec{w} = X^T\vec{r}$, where X is a matrix of all samples $\vec{x}_i$, $\vec{w}$ is a vector of the weights, and $\vec{r}$ is a vector of the real outputs. Solving for the weights becomes as simples as

$$\vec{w} = (X^TX)^{-1}X^T\vec{r}$$

Now the approximation can be calculated by $g(\vec{x}_i) = \vec{x}_i\vec{w}$.[Alp14]

This method can also be extended simply for univariate and multivariate polynomial regression. For the univariate case, given an order k for the function, set the variables as $x_1 = x, x_2 = x^2, ..., x_k = x^k$ and solve the same way as before. For the multivariate case, the cross products of variables are also considered as in the function $f(x_1, x_2) = w_0+w_1x_1+w_2x_2+w_3x_1x_2+w_4x_1^2+w_5x_2^2$, causing the transformation of variables as $x_1 = x_1, x_2 = x_2, x_3 = x_1x_2, x_4 = x_1^2, x_5 = x_2^2$. Again, the final solution for the weights and approximation is the same as in the linear case.[Alp14]

4 Results

In order to test the performance of the classification methods, the data must be divided into a training and testing set. The training set is used to create the model, and the testing set is used to evaluate the learned performance. Unfortunately, many models suffer because the training sample is not infinite, so the distributions are not accurately estimated. This leads to problems with overfitting, where the model works well on the training data but poorly on the testing data.

Cross validation attempts a solution to this problem by using the data to create multiple, randomly inter-selected sets of training and testing data. With m-fold cross validation, the data is broken into m samples of equal size for testing, with the rest of the data not selected for testing being used for training. This method provides more realistic accuracy numbers for the data, and lessens the effects of overfitting. Five fold cross validation has been used for all experiments on the data set. No hyperparameter adjustment was needed for this machine learning method, and so no validation data split was used.

4.1 Accuracy Measures

For the experiments two measures of accuracy are used. The coefficient of determination R^2 provides a measure of model proficiency as the percentage of the variability between the independent and dependent variables that can be explained. More practically, R^2 serves as a comparison between a horizontal line and a chosen model. Better performance is indicated by R^2 from zero to one, while R^2 is less than zero for worse performance. Formally,

$$R^2 = 1 - \frac{\sum_i (y_i - f_i)^2}{\sum_i (y_i - \bar{y})^2}$$

where $\bar{y}$ is the mean of the i y_i-values to model, and f_i indicates the associated prediction of the model.

The root mean square error provides a measure of how concentrated the data is around the model. More practically, it is the standard deviation of the residuals. Better performance is indicated by a lower value, with zero being the best possible performance. Formally,

$$RMSE = \sqrt{\sum_i (f_i - y_i)^2}$$

where again y_i are the real values and f_i are the predicted values.

4.2 Variable Contribution

In order to gain further understanding of the variables individual contributions to the model, multiple metrics are used. Assuming the null hypothesis, the p-value is the probability that a more improbable or extreme result would be obtained. This measure is used only to determine statistical significance as a boolean, because, "Smaller p-values do not necessarily imply the presence of larger or more important effects, and larger p-values do not imply a lack of importance or even lack of effect" [WL16]. For the automobile variables, all p-values were well below the standard of 0.05, indicating that all contributed in a statistically significant way to the regression.

Mutual information shows the dependence of two variables by measuring the information or entropy obtained about one variable through the other variable [CT91]. This measure of dependence also fits with our error measurements, since, "under some reasonable assumptions, features selected with the mutual information criterion are the ones minimising the mean squared error and

the mean absolute error." However, "the mutual information criterion can fail in selecting optimal features in some situations," which is why multiple measure are used [FDV13].

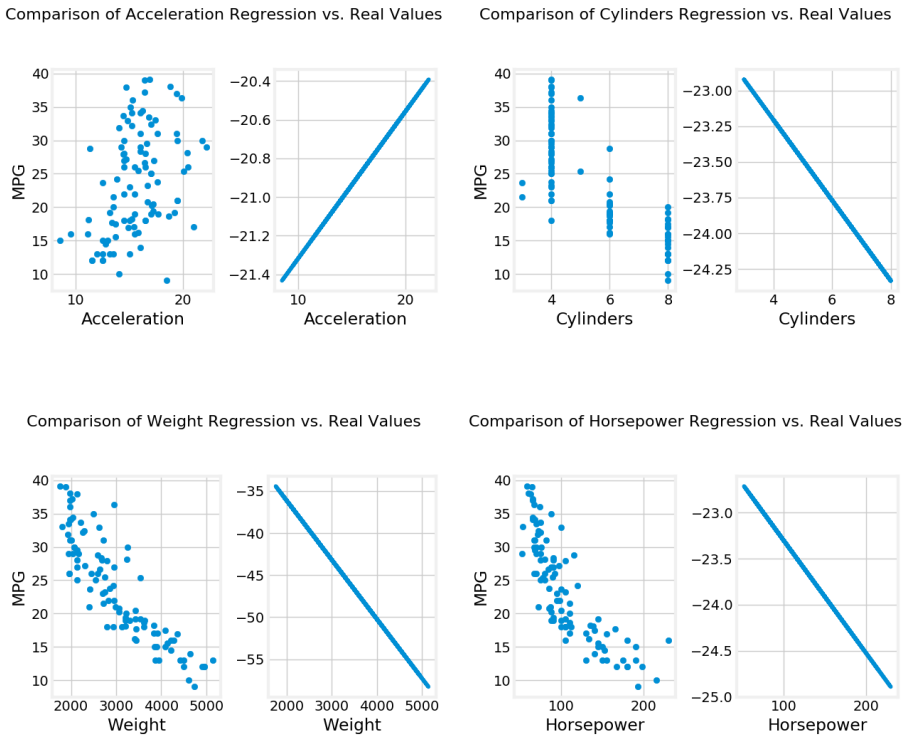
The F-test for regression value for each regressor is a ratio of the variance explained by the variable to the variance unexplained by the variable and functions as a measure of linear dependency. The F-test will not capture non-linear relationships, and so it is used in a comparison to other measures to explore which variables contribute in a linear way, and which contribute non-linearly.

The coefficient of determination R^2 is explored in two separate ways. First, the R^2 of each single variable is compared to all the others. Second, the ΔR^2 between the total accuracy and the accuracy with one variable removed is compared to all the others.

Feature	$p < 0.05$	Mutual Info	F-Test	Linear R^2	Linear ΔR^2
Cylinders	yes	0.75	0.68	0.60	0.01
Displacement	yes	0.99	0.82	0.64	0.03
Horsepower	yes	0.91	0.68	0.60	0.01
Weight	yes	1.0	1.0	0.69	0.48
Acceleration	yes	0.24	0.10	0.16	0.00
Model Year	yes	0.46	0.23	0.33	1.0
Origin	yes	0.29	0.21	0.31	0.11

Table 2: Individual variable significance and contribution. Mutual Information, F-Test, and ΔR^2 have been scaled by the maximum so that 1.0 represents the largest value.

From the p-value test it is apparent that the variables chosen are at least statistically significant. Looking at mutual information, F-test value, and R^2, it would appear that the top half of variables is most correlated with the gas mileage. Across these three tests, weight scored highest. However, when comparing ΔR^2, Model Year seems to contribute the majority, despite poor performance on earlier tests. Comparing this to intuition, it makes sense that as the weight of the car decreases and the model year increases, the gas mileage of a car would increase.



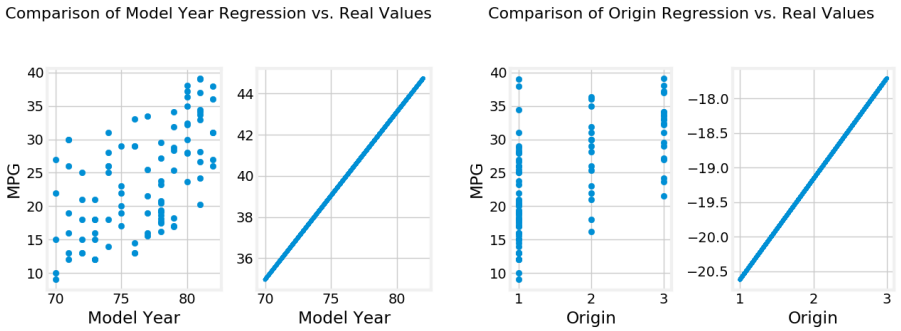


Figure 1: On the left of each figure is the real gas mileage value plotted against a certain variable, and on the right is the learned linear regression for that variable.

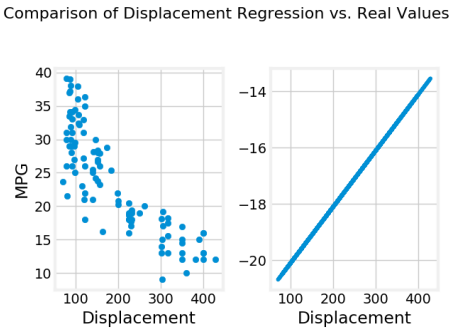


Figure 2: On the left of each figure is the real gas mileage value plotted against displacement, and on the right is the learned linear regression for displacement. It appears that the learned regression is directly opposite the data.

Above are graphs showing a comparison of each variable for the real gas mileage value and the learned regression. For most of these variables the graphs are very intuitive, however the graph for displacement appears to make little sense, with the regression running directly opposite the real data apparent trend.

4.3 R^2 *Accuracy*

For all of these methods, accuracy refers to the R^2 value over five-fold cross validation, and is given in Table 3. In a comparison of all methods, the highest accuracy achieved was a tie between the basic and z-normalized quadratic regression, at 0.86. For this data, the linear results also performed well, achieving 0.81; however, the cubic regression was significantly worse than a horizontal line regression. From these results it makes sense that some of the variables are slightly quadratic, something that can be seen in Figure 1 for the weight, horsepower, and displacement variables.

Normalization had a negligible effect on both the root mean square error and R^2 value for all of the regressions attempted. Only in combination with a reduction in dimension did the accuracy change. The most interesting change from using principal component analysis can be seen in Table 3. The accuracy of the cubic regression actually improves with dimension reduction, maxing out at 0.80 with four basis vectors. For all other methods, the accuracy decreases as the dimensionality is reduced.

Order	Basic	Normalized	PCA (6)	PCA (5)	PCA (4)	PCA (3)	PCA (2)	PCA (1)
Linear	0.81	0.81	0.68	0.60	0.60	0.61	0.50	0.51
Quadratic	0.86	0.86	0.74	0.67	0.68	0.63	0.53	0.50
Cubic	-	-	0.63	0.78	0.80	0.66	0.62	0.51

Table 3: R^2 accuracies for the original features, normalized features, and the features reduced using principal component analysis. Dashes represent negative (very low performance) R^2 values.

As the number of dimensions is reduced the runtime of the regression decreases, although the simplicity of the method and data size for this problem made the difference negligible. Table 4 conveys these results.

	PCA(7)	PCA(6)	PCA(5)	PCA (4)	PCA (3)	PCA (2)	PCA (1)
milliseconds	50	38	28	22	17	13	10

Table 4: Runtimes in milliseconds for running linear, quadratic, and cubic regression in series using reduced dimensions.

5 Summary

Z-normalization, principal component analysis, and multivariate polynomial regression methods were implemented and variable investigation was performed. Each method was applied to the problem of predicting gas mileage given automobile data. Investigation was done to determine the relative importance of different variables to the trend. Overall, an R^2 value of 0.86 was achieved using multivariate quadratic regression using no normalization or dimensionality reduction. It was determined that weight was the most significant contributer to this accuracy. It was also found that reducing the dimension using principal component analysis could reduce the runtime for all methods, but was unnecessary for this scope.

The objectives of this project were to both investigate a basic regression method and to practice investigating and evaluating a machine learning method and problem. From this I learned how to implement regression across multiple orders. More importantly, I discovered much more about variable selection and comparison methods for regression, including mutual information, p-values, and the F-test. I also practiced selecting an error function for a new problem, which was more difficult and complicated than a simple classification accuracy. In the future, more data collection and more variables could make this problem space more interesting or more accurate, but most likely better methods for prediction besides multivariate polynomial regression would be appropriate.

References

[Alp14] Ethem Alpaydin. *Introduction to Machine Learning Third Edition.* The MIT Press, 2014.

[cmu] http://lib.stat.cmu.edu/datasets/.

[CT91] T.M. Cover and J.A Thomas. *Elements of Information Theory.* John Wiley and Sons, 1991.

[DHS01] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification Second Edition.* Wiley, 2001.

[FDV13] Benoît Frénay, Gauthier Doquire, and Michel Verleysen. Is mutual information adequate for feature selection in regression? *In Neural Networks*, 48:1–7, 2013. https://doi.org/10.1016/j.neunet.2013.07.003.

[WL16] Ronald L. Wasserstein and Nicole A. Lazar. The asa's statement on p-values: Context, process, and purpose. *The American Statistician*, 70(2):129–133, 2016. http://www.tandfonline.com/doi/pdf/10.1080/00031305.2016.1154108.

6 Appendix

The code for this project can be found on GitHub at https://github.com/LambentLight/poly-regression. Please contact me with any questions or considerations.