Homework 9

Jiachen (Chris) Lu —— jlu1211@uw.edu

December 8, 2022

---

1 Introduction and Overview of the Question

Our goal is to fit various types of models to a data set of fuel consumption (mpg) of various cars as a function of different attributes such as engine displacement, horsepower, and weight. We will investigate the quality of our constructed models using QR decomposition and compare whether higher complexity results in better model performance or not. We will use QR decomposition instead of any other least square methods, such as LU decomposition and SVD. Due to stability and cost of solving, we prefer QR decomposition in this report. We will investigate them later in the report. We will also relate our observation to numerical stability and conditioning of the problem, so we can make conclusion of which model has the best performance.

The provided files contain training data sets: X_{train} and Y_{train}. X_{train} is a matrix of size 254 x 3 that each column contains engine displacements, horsepower, curb weight of the vehicles, respectively. Y_{train} contains the mpgs. The test data sets X_{test} and Y_{test} are analogous to the training set, but contains less instances - 138 instances. We will use the test data set to examine quality of our model we constructed with training data set.

We will first find a function $y(x)$ that is only a function of one of the features (Displacement/Horsepower/Weight) and denote the mpg of that car. For every function we define, we will formulate appropriate least squares problems using the training data set to find the parameters θ_j that define the models. We will use least squares to minimize the error between the models and test data. Then we will find the relative error of each model using test data sets. We will perform the same task for higher order analogues for the one feature function $y(x)$ to observe whether higher order will improve the performance. Next, we will consider models that depend on multiple features at a time instead of one feature. We will use multiple features to test whether adding more features will improve the performance. We will formulate appropriate least squares problems to find θ_j again and find relative errors on the test set to compare it to the previous models. Our goal is to find which model has the best performance.

2 Theoretical Background and Description of Algorithms

2.1 MPG as a function of displacement/horsepower/weight: One Feature Model

We will construct function y(x_i) that denote the mpg of a car with one specific feature (x_i): displacement, horsepower, weight. We are constructing y(x_i) to find the parameters θ_j that define the models. Also, we will be using the model to plug in test data sets to compute relative error to the original data. Equation are shown below. i represents index of which feature: i = 1 (displacement), i = 2 (horsepower), i = 3 (weight). K represents K-th degree models.

$$y^{(i,K)}(x_i) = \theta_0 + \sum_{k=1}^{K} \theta_k x_i^k \tag{1}$$

2.2 MPG as a function of displacement/horsepower/weight: Multiple Feature Model

We will construct y(x) that denote the mpg of a car with multiple features: displacement, horsepower, weight. We construct mpg function with multiple feature model to test whether it improves the estimation or not. i, j, k represents index of feature.

$$y^{(i,j)}(x) = \theta_0 + \theta_1 x_i + \theta_2 x_j + \theta_3 x_i x_j + \theta_4 x_i^2 + \theta_5 x_j^2 \tag{2}$$

$$y^{(i,j,k)}(x) = \theta_0 + \theta_1 x_i + \theta_2 x_j + \theta_3 x_k + \theta_4 x_i^2 + \theta_5 x_j^2 + \theta_6 x_k^2 \tag{3}$$

2.3 Least Square Problem Solver

The QR factorization allows us to express a matrix having linearly independent columns as the product of a matrix Q having orthonormal columns and an upper triangular matrix R. If Matrix A is a rectangular matrix that is m x n with m≥n, the bottom (m-n) rows of an mxn upper triangular matrix consist zeros.

To compute matrix Q, we will take orthogonal set of column vectors of matrix A and orthonormalize it. Then we will compute matrix R using following property: $QQ^T = I$. QR factorization is essential to obtain solution to the linear system.

$$A = QR \rightarrow Q^T A = Q^T QR \tag{4}$$

$$Q^T A = IR \rightarrow R = Q^T A \tag{5}$$

Given matrix A - n by m size - SVD decomposes the matrix A into 3 matrices: U, Σ, and V. Matrix U and V are real unitary orthogonal matrices which have the property that when the matrix multiplied by its transpose, the product is identity matrix. Matrix Σ is non-zero entries diagonal matrix in descending order.

$$A = U\Sigma V^T \tag{6}$$

$$UU^T = I \;\; \& \;\; VV^T = I \tag{7}$$

In this report, we chose QR factorization over SVD to solve for θ_j because QR has better numerical stability. In this report, we performed QR factorization by the Householder Reflection instead of Gram Schmidt. When we use SVD, we make assumption that sigma matrix is invertible which significantly decreases numerical stability. Difference between performance of SVD and QR factorization is caused by how we decompose the column space of matrix A. If we use Householder reflection, we will have two possible projections to take at each step which significantly improves the numerical accuracy of the method.

2.4 Relative Error

Relative error is used to calculate the quality difference between the original matrix and low rank approximation on its matrix. In this report, it is used to compare the output, so we can determine which model has the best performance. Calculation method is the following.

$$RelativeError(i) = \frac{1}{||Y_{test}||_2^2} \sum_{k=1}^{138} |y^i((X_{test})_{ji}) - (Y_{test})_j|^2 \tag{8}$$

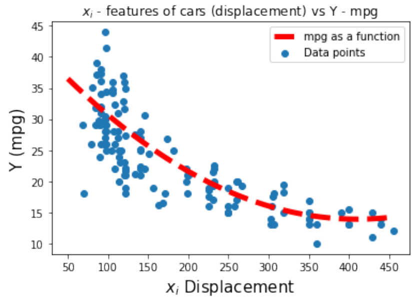
3 Computational Results

Relative Error: Single Feature				
Features	K = 2	K = 4	K = 8	K = 12
Displacement	0.0340543	0.0339045	0.0328288	0.0328732
Horsepower	0.0382639	0.0391719	0.0389401	0.0623139
Weight	0.0287830	0.0287925	0.0297802	0.0304722

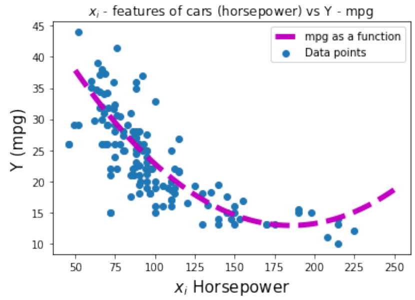
Table 1: Relative Error of Single Feature with different K values.

Relative Error: Multiple Features	
$y^{(1,2)}$: Displacement & Horsepower	0.0306935
$y^{(1,3)}$: Displacement & Weight	0.0290925
$y^{(2,3)}$: Horsepower & Weight	0.0306848
$y^{(1,2,3)}$: Displacement & Horsepower & Weight	0.0301054

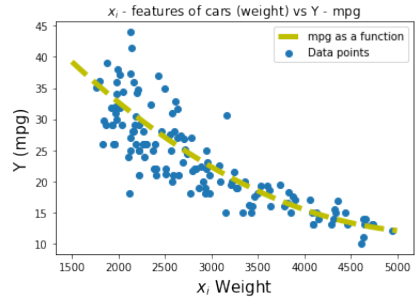
Table 2: Relative Error of Multiple Features.



(a) x_i (displacement) vs Y (mpg)

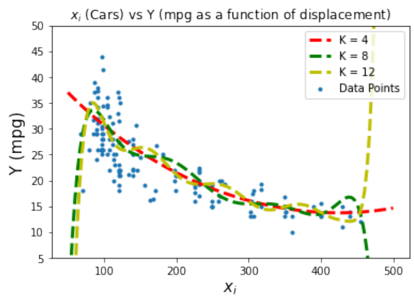


(b) x_i (horsepower) vs Y (mpg)

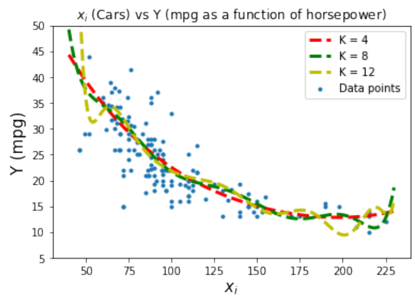


(c) x_i (horsepower) vs Y (mpg)

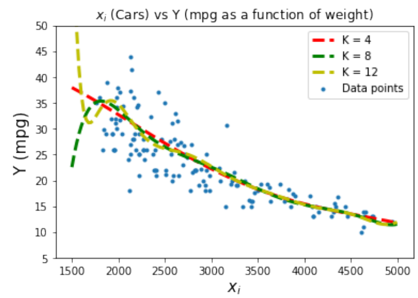
Figure 1: plots of 2D scatter plot of x_i vs Y of the test data overlaid with a line plot of model $y^{(i,2)}$ (1)



(a) x_i (Cars) vs Y (mpg: displacement)

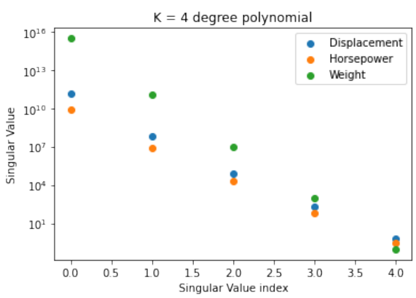


(b) x_i (Cars) vs Y (mpg: horsepower)

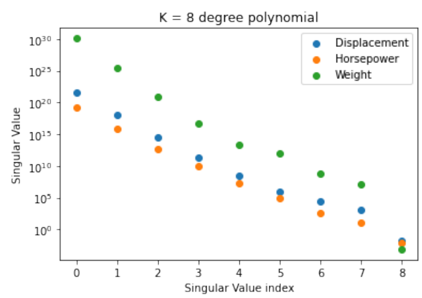


(c) x_i (horsepower) vs Y (mpg: weight)

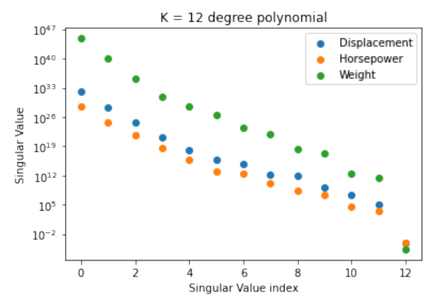
Figure 2: plots of requisite scatter and line plots of different K-th degree showing the behavior of displacement/horsepower/weight model respectively



(a) K = 4th degree



(b) K = 8th degree



(c) K = 12th degree

Figure 3: plots of Singular Values Index vs Singular Values of each feature (displacement/horsepower/weight) with different K degrees

Condition Number: K = 2	
Displacement	353554.86930919316
Horsepower	132730.3689150394
Weight	157867713.65878704

Table 3: Condition number of each feature at K = 2: Measure of how well/ill conditioned

Our first step is to load the given data set. Next, we will create Vandermonde Matrix with respect to specific feature we want to investigate. Benefit of creating Vandermonde matrix is that it allows us to create the matrix in ascending power order. The first column is power to zero, and second column is power to one, and so on. Then, we will use QR decomposition to find the parameters θ_j that define the model that is different for each model. We will use QR instead of other least square solver because we are formulating θ_j to construct the model y(x) that predicts the mpg of certain cars accurately (1). According to Table 3, using SVD will cause large condition number which will affect the result. Instead, we will use QR decomposition which is more stable numerically. After creating prediction model, we will plug in given test data into the model to determine relative error (8). We are going to calculate relative error to observe the quality of our model and compare with other model.

In Table 1, we can see among those three features, displacement, horsepower, and weight, weight has the smallest relative error, then it is displacement, and lastly horsepower. This is true for any K-th degree polynomial. As we can see in table 1, weight is the better predictor of the mpg than other two features. Since the error is small, we can barely distinguish which feature is better predictor. In fact, in figure 1, three plots with the line plot of model $y^{(i,2)}$ (1) capturing the data points cannot tell which one captures the data points better.

Next, we performed higher order analogues of the model. More precisely, K-th degree models of the form (1) for each feature. We performed the same task for higher order analogues model as what we did in figure 1. As we can see in Figure 2, when x_i increases, Y (mpg) decreases for all three features with three different K values. In Figure 2 (a), when K = 4, 8, and 12, line plots are relatively close to each other. For Figure 2 (b) and (c), we can also say line plots are relatively close to each other with different K value. In fact, in Table 2, difference between K value for displacement feature is approximately 1×10^{-3} that we can consider it very small. Although it is interesting to note that as K value increase, it does not necessary correspond that relative error will decrease. Evidently, when we observe Horsepower and Weight feature, relative error increases by 1×10^{-3} as K increases. However, it is still considered very small.

To examine the relative error more closely, we explored whether there is a point of diminishing returns in terms of degree K. To investigate the diminishing returns, we took a look at condition number of the matrices in our least square problem. We performed SVD (7) on matrix A which we obtained by creating Vandermonde matrix of specific feature (Displacement/Horsepower/Weight). Then, we obtain matrix Σ that contains singular values in descending order. Figure 3 shows the singular values of different K value with different features. For all three plots in figure 3, singular value is decreasing as singular value index decreases. When we examine the point of diminishing returns, we look at the singular value where it is the smallest. Therefore, the end of singular value index is the point of diminishing returns for all three features. It is approximately 10^0, 10^{-1}, and 10^{-3} for K = 4, 8, and 12 respectively. We can say at those points, increasing K gives a decreased marginal gain in the overall quality of the model.

Next, we examined models that depend on multiple features at a time by constructing new y(x). For two feature model, we used equation (2), and for all three feature model, we used equation (3). To formulate the appropriate least squares problem to find the θ vector for these models using the training data, we constructed matrix A by putting corresponding feature training data into the matrix. Then we performed QR decomposition to determine the θ vector. After getting θ, we multiplied by test data to get predicted y values, then calculated relative error for each model. In Table 1, we can see that relative error with multiple features in general have better performance than relative error with single feature (table 1). Relative error with three features do not have the best performance compared to other models. In fact, the best performance model is displacement and weight model with 0.0290925, which is 1×10^{-3} difference. However, adding features will definitely improve the performance of the model.

As a result, according to relative errors in table 1 and table 2, the model that performs best among the many models that we trained is single feature relative error model with weight at K = 4 which has 0.0287830 relative error.

4 Summary and Conclusions

Fitting various types of models to a data set of fuel consumption (mpg) of various cars as a function of different attributes using QR decomposition allows us to visualize which model has better performance in numerical stability and conditioning than other models. We saw the trend of increasing in x_i decreases Y (mpg) which indicate that later indexed cars have better fuel consumption than early indexed cars for different features (Figure 1, 2).

In table 1 and 2, we saw the relative error with single feature and multiple features. Increase in feature will improve the performance which was indicated by comparing K = 2 displacement, horsepower, and weight relative error in table 1 to relative errors in table 2. It is interesting to find that the model that depend on more features have better approximation than single or fewer feature models. This result can infer that when we are computing least square problems, incorporating more features can produce better estimation results. However, there is potential caveats or concerns when we use more features. In fact, relative error with three feature model is greater than model with two features and model with only weight feature (table 1, 2). This could be caused by condition number of matrices. According to Table 3, both displacement and horsepower feature model has 10^5 condition number while weight feature has 10^8. Despite decrease in marginal gain in singular values shown in figure 3, overall quality of the model is not reduced. However, relative error is increased due to condition number which is caused by how sensitive the answer is to perturbations in the input data and to roundoff errors made during the solution process.

Nevertheless, we have sustained the relative error small because we used QR decomposition. Unlike SVD where computers cannot capture certain value greater than 1×10^{16} (figure 3), QR decomposition uses Householder Reflection to improve numerical stability. In addition, we are also concerned about point of diminishing return in terms of the degree K. According to figure 3, the greater the K value becomes, less singular value it will become, which means we will extract less singular value from greater Singular Value index.

For possible future direction of this work, we can perform least square problem to other data set to examine numerical stability and quality improvement on higher complexity models. Unlike only using three features, we can use large number of features to construct multiple features model to predict the relative error.

5 Reference

Stack Overflow (https://stackoverflow.com)
Wikipedia (https://en.wikipedia.org/)
EECS Berkley (https://inst.eecs.berkeley.edu/ ee127/sp21/livebook/l_svd_lineqs.html.)
Michael Rozman, D. of P. (n.d.) (https://www.phys.uconn.edu/ rozman/Courses/m3511˙18s/downloads/)