Student Name: Minh Phan
Course: AMATH 352
Email address: minhphan@uw.edu

Homework 9

---

1 Introduction and overview of the problem

In numerical analysis, one of the most important tasks while creating models to use on data is evaluation. Checking the quality of models is crucial in ensuring that our models work against different required range of data set out for the model production itself, and whether they are efficient in practice. Some of the topics in model evaluation are numerical stability and conditioning, which will be demonstrated in this report. From the class lecture 27 and 28, we already know that conditioning (or a condition number) refers to the how much change resulted in the model output if one is to add a small change into the input. If the change is substantial, then it shows that the model lacks stability and is ill-conditioned, and well-conditioned otherwise. In this report, I have tested different least squares problem models using some and all features of the dataset.

2 Theoretical background and description of algorithm

In this report I build my least square regression models using the underlying QR decomposition algorithm for my models as both Singular Value Decomposition (SVD) and QR Decomposition functions are programmed and fine-tuned to deliver high reliability and fast speed. It has been known that SVD are more reliable for rank-deficient matrices [1]; however, the size of the dataset is not big enough to show a significant quality difference betewen the two algorithms, so this compensation may be acceptable. A rough code script on the implementations of SVD in some of the models is attached in the code files (but commented out) submitted along with this report; however, there are big relative errors in these implementations as they are not as optimized, and henceforth will not be mentioned in this report.

Since this report focuses on the QR decomposition algorithm, a broad background on it should be addressed.

According to what is defined in class lecture 17, computing QR factorization (or decomposition) on a matrix A is analogous to performing Gram-Schmidt process on the columns of A. We create an orthonormal matrix Q from new orthonormal vectors obtained from the Gram-Schmidt process on the columns. The other matrix R includes entries that when multiplies with Q will reproduce the original matrix A. The classical Gram-Schmidt process is notoriously known to be unstable [3], but this instability is reduced for the modified Gram-Schmidt process used in NumPy's implementation of the process in Python.

Before diving into the report, here are some definitions that may be useful to know to better understand what's happening:

- Relative error

  Relative error or approximation error in a data value is the "ratio of the absolute error of a measurement to the measurement being taken"; in other words, it is relative to the real value of the value of the data point being referenced [5]. In this report, the formula for the relative report is defined as:

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

  where i is the order of the feature column in the dataset (1, 2, or 3), and $||Y_{test}||_2$ is the 2-norm of the label vector Y_{test}.

- Condition number

  A condition number of a problem measures the sensitivyty of the solution to small pertubations of the data. It varies by the problem and the input data, by the norm that is used to measure the size and what pertubations are measured on[6]. If a problem is ill conditioned, then the condition number is large; however, there is no exact definition on what counts as 'small' and 'large'[4]. In least square problems, especially the ones shown in this report, condition numbers are calculated from the R matrix factorized from QR decomposition [2].

3 Computational Results

The dataset has 3 features, namely displacement, horespower, and weight, and 1 label mpg which represents fuel consumption. Name the three features' individual data points x_i, with $i = 1, 2, 3$, respectively. The models used in the first two tables and two figures are configured by the form:

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

The matrix form of these models can be achieved using Vandermonde matrices.

Feature	displacement	horsepower	weight
Relative error	0.0341	0.0383	0.0288

Table 1: Relative error for predictions on test dataset for each feature model, K=2

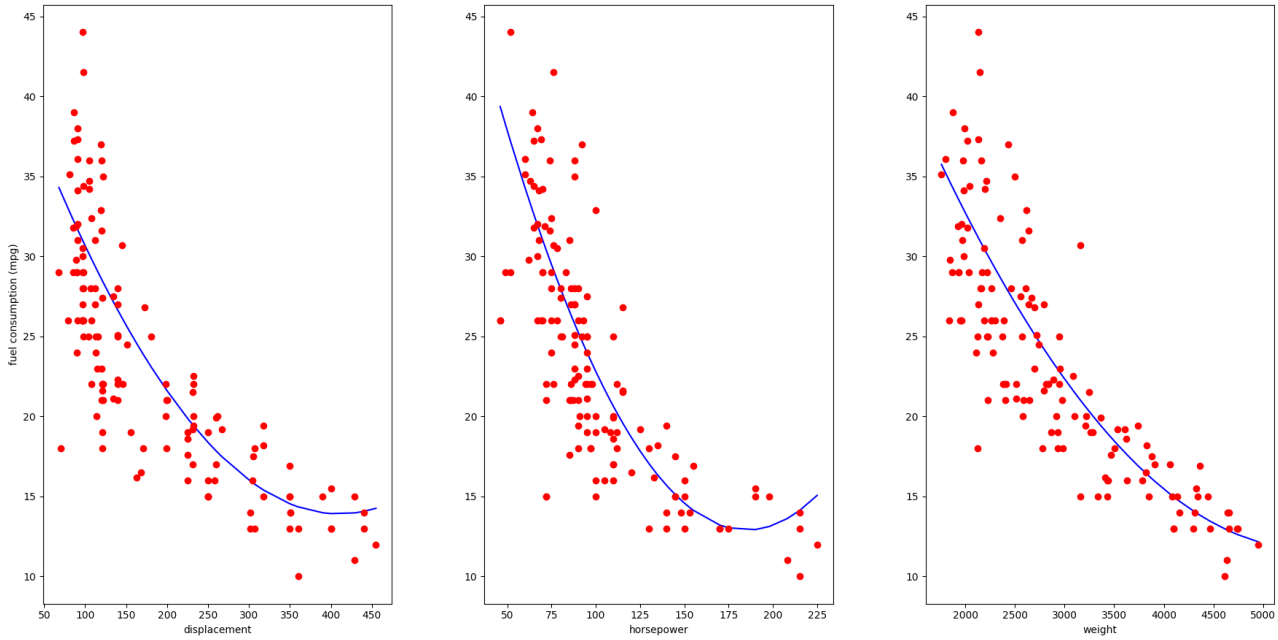

Figure 1: (From left to right) fuel consumption (mpg) on displacement, horsepower, and weight observation points and their model's line regressions with K=2, respectively.

Observing the shapes of the regression and the actual observations in Figure 1, as well as the relative errors of each regression reported in Table 1, it looks like the weight feature is the best predictor of the mpg out of three. Unlike the other two features' model regressions, the shape of the weight feature's is not curved up at any point, which practically makes more sense, because added weight should not efficiently tune up the engine by any chance.

Relative error			
Feature	displacement	horsepower	weight
K=4	0.0339	0.0392	0.0288
K=8	0.0328	0.0389	0.0298
K=12	0.0329	0.0623	0.0305

Table 2: Relative error for predictions on test dataset for each feature model, for K=4, 8, and 12

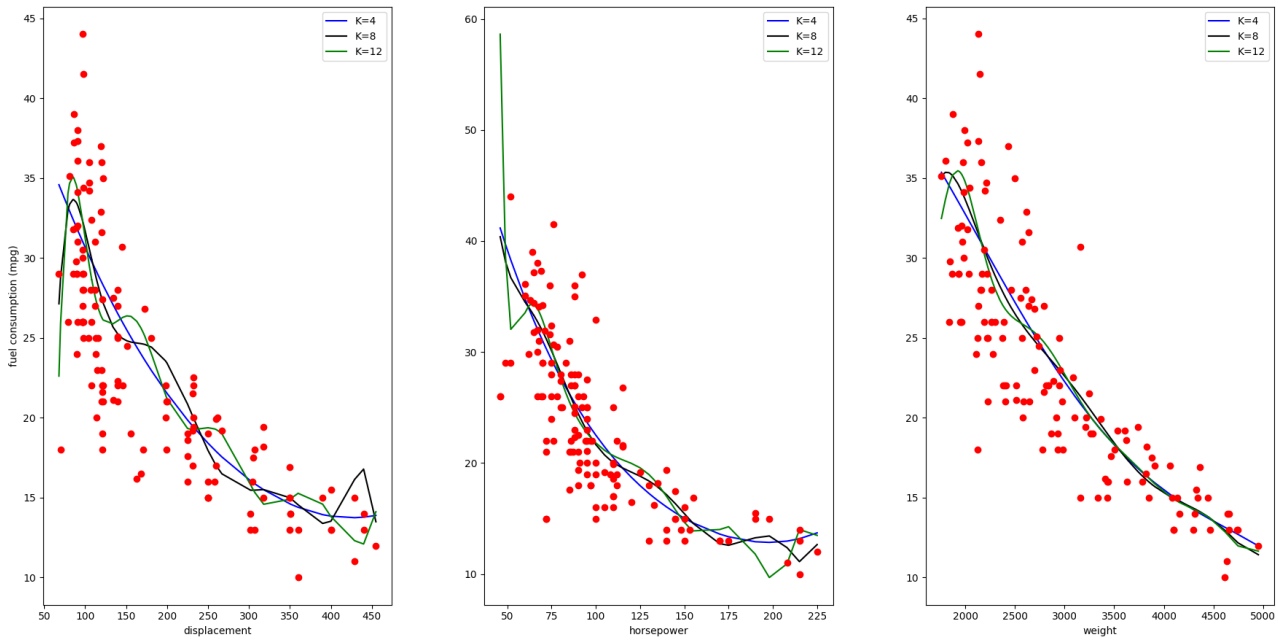

Figure 2: (From left to right) fuel consumption (mpg) on displacement, horsepower, and weight observation points and their model's line regressions with, respectively.

Condition number			
Feature	displacement	horsepower	weight
K=2	3.536×10^{5}	1.327×10^{5}	1.579×10^{8}
K=4	2.395×10^{11}	2.810×10^{10}	3.167×10^{16}
K=8	2.676×10^{23}	1.698×10^{21}	1.544×10^{33}
K=12	7.721×10^{35}	4.860×10^{32}	2.012×10^{50}

Table 3: Condition number for predictions on train dataset for each feature model, for K=2, 4, 8, and 12

Observing relative errors results on Table 2 and the graphs in Figure 2, there is indeed a point of diminishing returns in terms of the degree K. For the models concerning the displacement feature, the relative error is at its lower at K = 8. For the horsepower and weight features, both are K=2. We can see that the larger the K value (the higher the order of the polynomial functions), the rougher and more oscillating and unstable the regression lines are, and at some point they start overfitting the data points and do not accurately represent the general trend anymore.

Considering the table 3 of condition numbers, we can see that the larger the K values, the larger the condition number in all of the three features' models. This table uses data on training dataset, but the trend is also expected if using the test dataset. This information is helpful in choosing the best model among those that have minimal relative error differences.

The multi-feature models used in relative error calculations in Table 3 below are:

$$y^{12}(\underline{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \theta_4 x_1^2 + \theta_5 x_2^2$$
$$y^{13}(\underline{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_3 + \theta_3 x_1 x_3 + \theta_4 x_1^2 + \theta_5 x_3^2$$
$$y^{23}(\underline{x}) = \theta_0 + \theta_1 x_2 + \theta_2 x_3 + \theta_3 x_2 x_3 + \theta_4 x_2^2 + \theta_5 x_3^2$$
$$y^{123}(\underline{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \theta_4 x_1^2 + \theta_5 x_2^2 + \theta_6 x_3^2$$

Model	y^{12}	y^{13}	y^{23}	y^{123}
Relative error	0.0307	0.0291	0.0307	0.0301
Condition number	7.125×10^5	2.423×10^8	1.630×10^8	2.014×10^8

Table 4: Relative error and condition number for each model described above

From table 4, we can see that adding features may help to improve the performance of some single-feature models, like in the case of y^{13} and y^{12} models to the displacement and horsepower feature models. However, this is not the case for the weight feature model, as combining other features seem to reduce its accuracy (relative error increases), though not too much.

To pick out the model that performs the best among these models, one can easily pick out the weight feature model at $K = 2$ due to its substantially small condition number comparing to the models at K=4, 8, 12, and the multi-feature models. It also consistently has the lowest error compared to the single-features' models. This decision, however, should be up to further decision. Should weight be the only factor that matters to fuel consumption rate (mpg), or the other features may play some role, albeit rather smaller, in this relationship?

4 Summary and Conclusions

This report has shown some statistics and visualizations served to evaluate linear regression models between different features, namely displacement, horsepower, and weight and fuel consumption (mpg) of a dataset. The most applicable model with the lowest relative error (and highest accuracy) is a second-order polynomial function based on the weight feature, with $K = 2$: $y^{3,2}(x_3) = \theta_0 + \theta_1 x_3 + \theta_2 x_3^2$. This result comes with the assumption that QR decomposition is the underlying algorithm used to evaluate these models. Since the matrix used in this model is not row-deficient, which can be checked using Numpy's numpy.linalg.rank() function, it is safe to use QR decomposition in this case. However, this may not be the case for some of the models used in this report. Further testing and investigation of these models, as well as experimentation of other possible models are necessary to find the best possible one to use on this dataset.

References

[1] cmk (https://math.stackexchange.com/users/671645/cmk). *When solving a linear system, why SVD is preferred over QR to make the solution more stable?* Mathematics Stack Exchange. eprint: `https://math.stackexchange.com/q/3252377`. URL: `https://math.stackexchange.com/q/3252377`.

[2] David Bindel. *Week 5: Monday, Feb 27*. Cornell University, 2012.

[3] Wikipedia Contributors. *Gram–Schmidt process*. Wikipedia, Nov. 2022. URL: `https://en.wikipedia.org/wiki/Gram%5C%E2%5C%80%5C%93Schmidt_process#Numerical_stability` (visited on 12/10/2022).

[4] Stephanie Glen. *Ill-Conditioned Condition Number*. Statistics How To. URL: `https://www.statisticshowto.com/calculus-definitions/ill-conditioned-condition-number/` (visited on 12/11/2022).

[5] Stephanie Glen. *Relative Error: Definition, Formula, Examples*. Statistics How To, Nov. 2016. URL: `https://www.statisticshowto.com/relative-error/`.

[6] Nick Higham. *What Is a Condition Number?* Nick Higham, Mar. 2020. URL: `https://nhigham.com/2020/03/19/what-is-a-condition-number/`.