Data mining project report

- Predicting MPG For Cars

Jiannan Zhang

First section: motivation + description + data + algorithms.

1a. Project motivation and description:

Nowadays, the fuel economy is catching more and more attentions. Therefore, it is essential to have good estimates of fuel consumption for different types of cars. MPG, which stands for mile per gallon, is one of most important measures for that. In this project, I am trying to find the potential factors that affect the MPG and generate a regression model to analyze their relation. I would focus on seven predictor variables: number of cylinders, engine displacement, horsepower, weight of the vehicle, acceleration time, age of the vehicle and origin country.

1b. Data

The dataset is from the StatLib library, which is maintained at Carnegie Mellon University (The dataset was used in the 1983 American Statistical Association Exposition). Also, I found the details for each predictor online. Here are all seven predictors information:

1) Number of cylinders. This data is discrete, with three possible values, 4, 6 and 8. The more cylinders an engine has the more fuel it consumes but also provides more energy.

2) Normal engine displacement (cubic inches), which is a continuous variable and means the volume swept by all the pistons inside the cylinders of a reciprocating engine in a single movement from top to bottom. With larger displacement the engine will consume more fuel and provide more energy.

3) Mechanical horsepower (HP), which is a continuous variable and measures the power of engine. Although usually horsepower is positively related to the engine displacement some technology such as turbo will change this situation so we still assume the horsepower is an significant parameter in measuring fuel efficiency.

4) Weight of the vehicle (pounds). A heavy car will consume more fuel than a light car forging the same distance.

5) Acceleration time (seconds). This represents the average time the driver would need to

use to accelerate the vehicle from 0 to 80 miles/hour. Acceleration time is an indicator of the overall performance of the vehicle engine. Usually a car with larger horsepower and displacement will have a faster acceleration time but as mentioned above, some technology will deny this situation.

6) Age of the vehicle (years). How old the vehicle is possibly affects the fuel consumption of a vehicle. Since every car has its longevity, which is either measured in time or miles, and we are not able to make sure if a car has exceeded its longevity, we cannot say the older the car is the more fuel it will consume forging the same distance. However we do expect the more years over a car's longevity the less the fuel efficiency will be.

7) Origin of the vehicle. This variable represents the country of the vehicle producer. 1 = the US; 2 = Germany and 3 = Japan. Cars from different country will have different quality and target consumers. We expect that Japanese cars will be more efficient since they are usually light.

1c. Algorithms

a. Ridge regression:

Ridge regression is one of the more popular, albeit controversial, estimation procedures for combating multicollinearity. It penalizes the size of the regression coefficients by adding a degree of bias to the regression estimates; it also reduces the standard errors. If we have some large weights for the predictors, the system for the model is not stable. That is the reason why we want to use ridge regression.

Mathematical explanation:
Suppose Y is the value we try to predict, X is a big matrix containing the entire feature, W is the weights for the ridge regression model, lambda is the ridge parameter. Then, by math, we can get:

$$W = (X^T X + \text{lambda} * I)^{-1} * X^T Y$$

Then we want to minimize the following:

||Y – XW||^2 + lambda *（||W||^2） ('*' means multiply,'^' means raise to a power)

Here, lambda has several roles:

1. λ controls the size of the coefficients
2. λ controls amount of regularization
3. As λ ↓ 0, we obtain the least squares solution

Second section: execution + results.

General idea:

There are few missing values of features in some instances of the data, since we focus on predicting MPG and those few data will not influence the result, so I just delete them.

After modifying the dataset, we now have 390 instances and each of them has seven features. Then I do the ridge regression in MATLAB for various lambdas. My goal is to find the lambda that minimize the following expression:

||Y – XW||^2 + lambda*||W||^2 ('*' means multiply,'^' means raise to a power)

Let us call the result of above expression **'error'。**

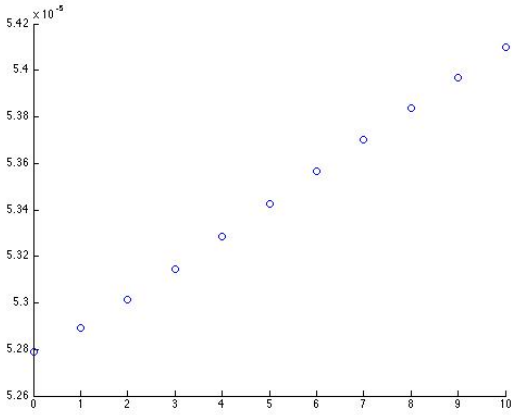
Another important implementation is that I use 10 – fold Cross-Validation to train and test the data. So each fold has 39 instances. Every time I use nine folds to train data and get the regularization parameter 'W'. Then I use that 'W' to test the data, so there will be 39 predicted value for that corresponding test fold. Then I find every difference between predicted value and true value and find mean of those difference and then add the regularization term, which is **lambda*||W||^2** to that mean. So this is the mean error for one fold out of ten-fold Crossing-Validation. Finally I find all the mean errors for ten folds and I take the mean again, denote that mean by lambda_error. Finally I store that error for the corresponding lambda in to an array. Now I plot the scatter plot against lambda_error and lambda and find the best lambda given a specific range.

Specific process:

I. Use all seven features from the dataset.
(This dataset is in the file named "Data")

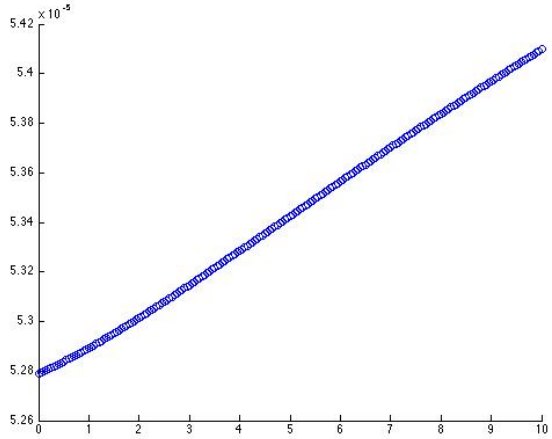
First I choose lambda from 0 to 10 with step = 1. I found that lambda = 1 is the best lambda for that. The data and plot is below:

```
lambda_err =
  1.0e-04 *
 Columns 1 through 9
   0.5279   0.5289   0.5301   0.5315   0.5329   0.5343   0.5356   0.5370   0.5384
 Columns 10 through 11
   0.5397   0.5410
```



Here comes something weird because the error increased all the time.

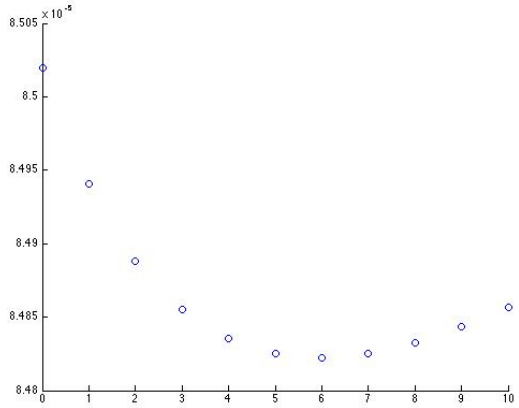
Then I make lambda from 1 to 10 with step 0.05 and again 0 is the best one. The plot is below:



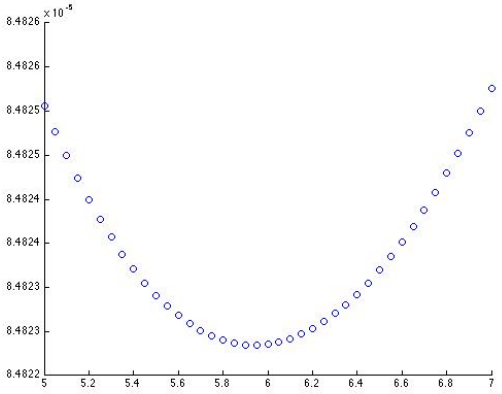
It is really surprising so far because the error keep increasing all the time. Now, I realized there should be something wrong. The reason comes out as I think through: There are three discrete variables, I should modify them use dummy variable, but actually after when I run the code several time and test it many times, I found only the variable that represents the year of the car influence the error (result) a lot. Then I found out that year actually has little influence on MPG by comparisons of errors and it has pretty small weights. I think the reason might be that the influence of year can be implicitly shown by cylinder and displacement because as year goes by, the technology makes the engine more efficient so that engine displacement goes down as year passed by. Finally I just delete that variable from my model.

II. Use modified data (6 features). This is model I
(This dataset is in the file named "Datamodify")

First I run it from lambda =0 to 10 with step = 1



so the best lambda is between 5 to 7, then I run it from 5 to 7 with step 0.05. The plot is below:



It shows that the best lambda is between 5.5 to 6.5. Then I run it with step = 0.1 and the plot and error is below:

lambda_err =

1.0e-04 *

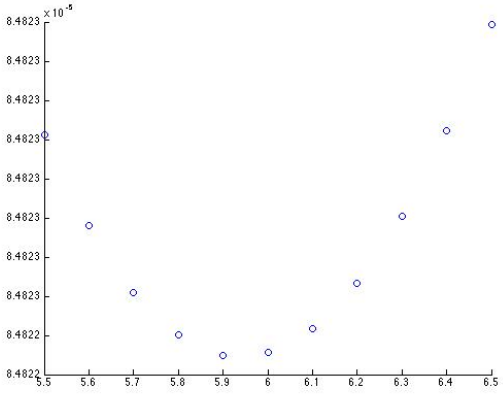
Columns 1 through 9

0.8482 0.8482 0.8482 0.8482 0.8482 0.8482 0.8482 0.8482 0.8482

Columns 10 through 11

0.8482 0.8482

Here are only four decimal places shown. They have same value when rounded.



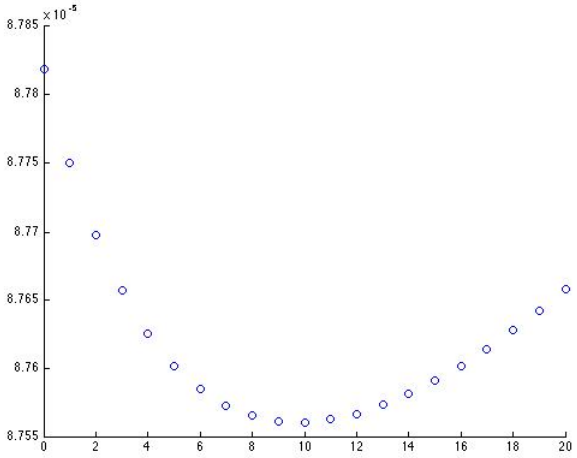
Now we find the best lambda to fit this model with **lambda = 5.9 and the corresponding error = 1.0e-04 * 0.8482.** The error is pretty small! This means this model is an excellent model.

The corresponding W(weights) for this model is W = [0.0873 -0.1021 0.0474 -0.2682 -0.5452 -0.0365 0.1033].

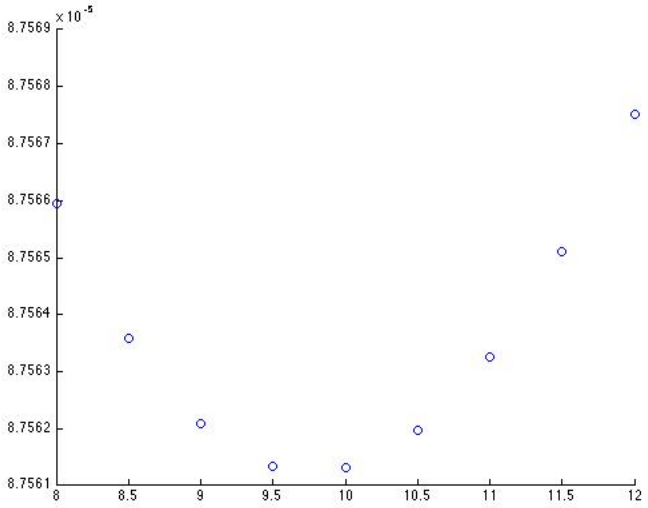
III. Data with all continuous variable (displacement, HP, weight, acceleration time). This is model II

(This dataset is in the file named by "Data2")

I want to know the relationship between all these continuous variable and the MPG. First I run it from lambda = 0 to 20 with step = 1. The plot is below:



it shows the best lambda is between 8 and 12, then I run it again from 8 to 12 and the plot is below:



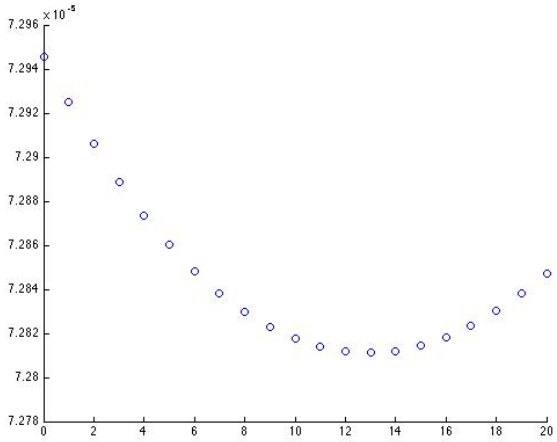
So the best lambda for this model is 10 and error is 1.0e-04 * 0.8756. Notice that this is a little big higher than error of previous one.

The corresponding W(weights) for this model is W =[0.0919 -0.0965 -0.2072 -0.5733 -0.0369]

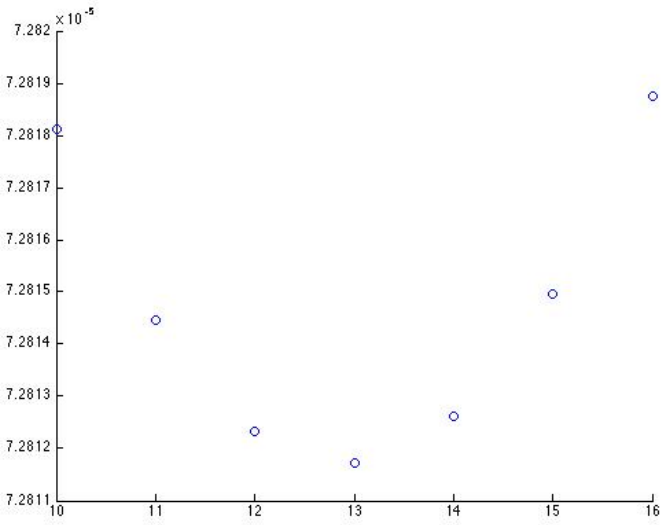
IV. Data with all discrete variable (number of cylinders, origin, years). This is model III

(This dataset is in the file named "Data3")

I am also interested in the relationship between all these discrete variable and the MPG. First I run it from lambda = 0 to 20 with step = 1. The plot is below:



this shows the best lambda is between 10 to 16. Then run it again from 10 to 16, the plot is below:



Now we find the best lambda is 16 and error is 1.0e-04 * 0.7281. Notice that this error is the smallest one in these three models

The corresponding W(weights) for this model is W = [-0.0468 -0.5582 2.2883 0.1355]

Third section: Discussion and analysis

In this project, basically I am trying to predict MPG by given some features of the car. After I finished the project, I found that the model is pretty good because it has very small errors even I combined variables with both continuous one and discrete one. In my project, I used the ridge regression algorithm, training and testing data by crossing-validaiton.

Since there are three models for different cases, it is reasonable to use different models in different care. For example, if you just have discrete predictors, you should use model III and it also has smallest error.

The weights for these model shows that horsepower and weight plays higher weight in regression, here is some sample weights for model I:

w = 0.0879 -0.1139 0.0075 **-0.2255** **-0.4932** -0.0800 0.0651
w = 0.0804 -0.1025 0.0120 **-0.2232** **-0.4859** 0.0118 0.1161
w = 0.0849 -0.0908 0.0096 **-0.2721** **-0.4812** -0.0237 0.1076

Here are some sample weights for model II:

W = 0.0913 -0.1071 **-0.2216** **-0.5395** -0.0297
W = 0.0923 -0.1243 **-0.2068** **-0.5246** -0.0657
W = 0.0936 -0.0841 **-0.2312** **-0.5807** -0.0408

All bold numbers are weights for horsepower and weight. It shows higher HP and weight will reduce the MPG a lot, which makes a lot sense to us. Also, notice the weight from model I, the weights for the last variable is for the variable origin, it shows it is positive related. Which means with higher origin, the MPG is higher. In our case, 1 = the US; 2 = Germany and 3 = Japan. So Japanese cars usually have the biggest MPG, then Germany, then U.S. in these three origins. It also makes lots of sense to us because we know it is usually true from our daily life experience. We can use this model to guide us to buy a car or use this model in the car industry to predict MPG before some cars come out and so on. Finally, data mining is fun and it explains our life and leads us to a better way.