Imperial College of Science, Technology and Medicine

Department of Mechanical Engineering

ME3 Statistics Coursework

Author:
Arohan Subramonia
CID:
01054062

Course Leader:
Dr. Nicola Fitz-Simon

March 13, 2018

Introduction

The fuel efficiencies of vehicles are dependent on a number of parameters. With the unique dataset given, the aim of this coursework is to construct linear regression models to aim to estimate fuel efficiencies based on a combination of these parameters. The analysis of this data is given in the following report.

1 Question 1

Exploratory Data Analysis

The following parameters were calculated for the fuel efficiency (litres/100km) data given:

Arithmetic Mean	4.7
Geometric Mean	4.6
Median	4.4
10% Trimmed mean	4.7
Arithmetic standard deviation	1.30
Geometric standard deviation	1.33

Table 1: Exploratory data analysis values

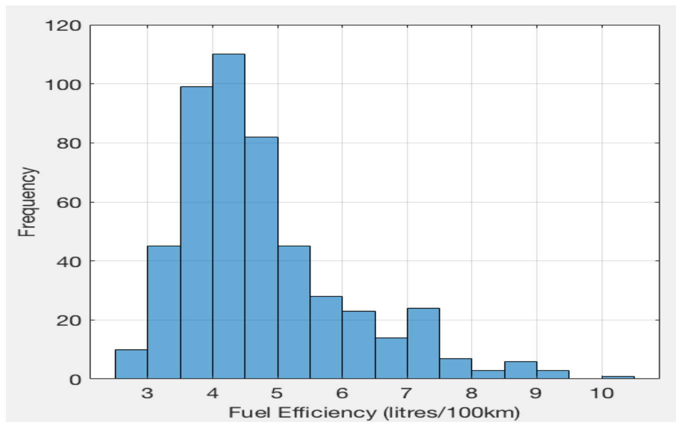

Figure 1: A histogram of the fuel efficiency distribution

The 10% trimmed mean is a better measure than the means or the median, as it is less affected by skewed data and outliers.

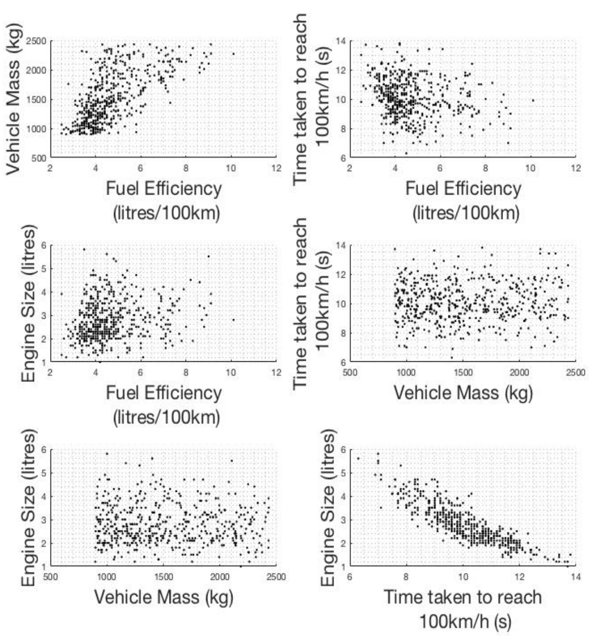

Figure 2: Scatter plots of all continuous variables

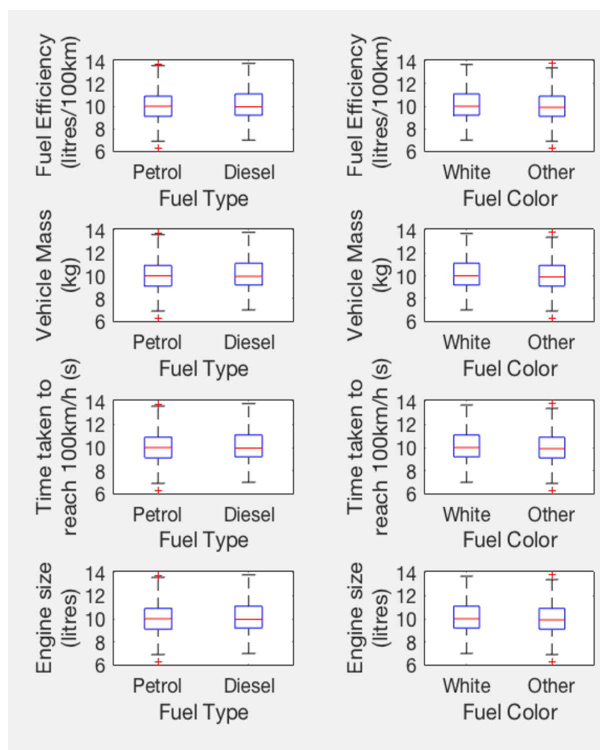

Figure 3: Boxplots of continuous by categoric variables

The scatter plots show that there may be some form of linear relationship between acceleration time and engine size; and potentially some relationship between vehicle mass and fuel efficiency. However, for the other pairs of data, there are not many easily discernible linear (or otherwise) relationships. To satisfy the assumptions of linearity and additivity, all continuous variables should be standardized by their arithmetic means and standard deviations. Therefore, the data was standardized in this way before moving onto the next section.

2 Question 2

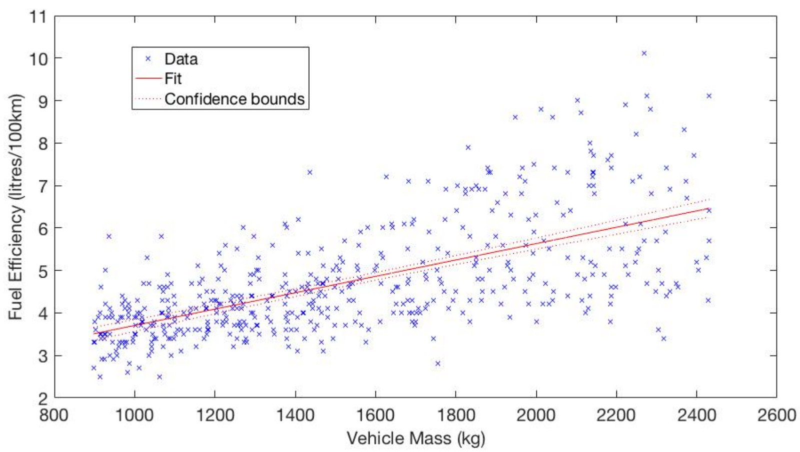

Figure 4: Scatter plot showing linear regression model of fuel efficiency in terms of mass.

This model is a passable initial linear approximation for fuel efficiency in terms of mass, as the two variables seem to have a positive relationship on the graph. This relationship could also be modeled by a quadratic relationship - however, this gives the exact same R squared estimate and larger values for the MSE and AIC.

Prove that R^2 is the square of the sample correlation coefficient $r(x,y)$:

$$\hat{y}_i = \hat{\alpha} + \hat{\beta} x_i \qquad \bar{y} = \hat{\alpha} + \hat{\beta}\bar{x} \tag{1}$$

$$R^2 = 1 - \frac{\Sigma_i^n (y_i - \hat{y}_i)^2}{\Sigma_i^n (y_i - \bar{y})^2} = \frac{\Sigma_i^n (y_i - \bar{y})^2 - \Sigma_i^n (y_i - \hat{y}_i)^2}{\Sigma_i^n (y_i - \bar{y})^2} \tag{2}$$

For simple linear regression, $\Sigma(y_i - \bar{y})^2 = \Sigma(y_i - \hat{y}_i)^2 + \Sigma(\hat{y}_i - \bar{y})^2$ which means:

$$R^2 = \frac{\Sigma(\hat{y}_i - \bar{y})^2}{\Sigma(y_i - \bar{y})^2} = \frac{\Sigma(\hat{\alpha} + \hat{\beta} x_i - \bar{y})^2}{\Sigma(y_i - \bar{y})^2} \tag{3}$$

From (1.b),

$$R^2 = \frac{\Sigma(\bar{y} - \hat{\beta}\bar{x} + \hat{\beta}x_i - \bar{y})^2}{\Sigma(y_i - \bar{y})^2} = \frac{\hat{\beta}^2\Sigma(x_i - \bar{x})^2}{\Sigma(y_i - \bar{y})^2} \tag{4}$$

$$R^2 = \frac{[\Sigma(x_i - \bar{x})(y_i - \bar{y}]^2\Sigma(x_i - \bar{x})^2}{[\Sigma(x_i - \bar{x})^2]^2\Sigma(y_i - \bar{y})^2} \tag{5}$$

Cancelling the $\Sigma(x_i - \bar{x})^2$ terms above, we get:

$$R^2 = \frac{[\Sigma(x_i - \bar{x})(y_i - \bar{y}]^2}{\Sigma(x_i - \bar{x})^2\Sigma(y_i - \bar{y})^2} = r(x,y)^2 \tag{6}$$

As $r(x,y) = \frac{\Sigma(x_i - \bar{x})(y_i - \bar{y}}{\sqrt[2]{\Sigma(x_i - \bar{x})^2\Sigma(y_i - \bar{y})^2}}$ which is the correlation coefficient.

Measures of accuracy - a comparison

The **R squared model** is very widely used, and a common predictor of the percentage variance of the model from the actual values.

The **(root) mean squared estimate** is a good parameter of a model as it gives an absolute value for the average residual or error of the data sample. By squaring the errors, we ensure that the mean error does not come out to be zero.

The **Akaike Information Criterion** is similar to the mean squared estimate, except it accounts for the number of parameters in the model, and therefore prevents over-fitting.

First we try and predict values of fuel efficiency with single parameter models for vehicle mass, acceleration time, and engine size. After obtaining R squared, MSE and AIC estimates for these, we try out linear and quadratic multivariate models with and without interactions, with the aim of maximizing the R squared value and minimizing the AIC value for each parameter. The overall results are plotted in the table below:

Type	Rsquared	MSE	AIC
'mass (linear)'	0.40815	0.59304	1159.7
'mass (quad)'	0.40815	0.59423	1161.7
'time (linear)'	0.043828	0.95809	1399.5
'time (quad)'	0.043843	0.96	1401.5
'engine size (linear)'	0.034128	0.96781	1404.6
'engine size (quad)'	0.039556	0.96431	1403.8
'Linear Multivariate'	0.65966	0.34379	891.04
'Pure Quadratic Multivariate'	0.66309	0.3424	895.97
'Linear Interactions'	0.76933	0.23782	716.55
'Quadratic Interactions'	0.77371	0.23476	716.96

Figure 5: Parameters for various models

```
Linear regression model:
    Nl100 ~ 1 + Nmass*Nt100 + Nmass*type + Nt100*type

Estimated Coefficients:
                    Estimate       SE         tStat       pValue
                   _________    ________    _______    __________

    (Intercept)     -0.46943    0.031081    -15.104    1.2352e-42
    Nmass            0.30604    0.030579     10.008    1.3695e-21
    Nt100            -0.1912    0.030667    -6.2348    9.7257e-10
    type             0.88475    0.043975      20.12    3.5955e-66
    Nmass:Nt100    -0.064931    0.021255    -3.0549     0.0023731
    Nmass:type       0.63732    0.044088     14.456    9.6989e-40
    Nt100:type      -0.12849     0.04396    -2.9228      0.003628

Number of observations: 500, Error degrees of freedom: 493
Root Mean Squared Error: 0.489
R-squared: 0.764,  Adjusted R-Squared 0.761
F-statistic vs. constant model: 266, p-value = 6.05e-151
```

Figure 6: Final linear model to predict fuel efficiency

The multivariate quadratic interactions model has 21 terms, which should be refined. Therefore, smallest (absolute) coefficients have been taken out, and using the 'step' function in MATLAB, coefficients are shuffled around to minimize the AIC value of the model. A trade off is made between number of coefficients to involve and R squared value. The final model is given in Figure 6 above.

3 Question 3

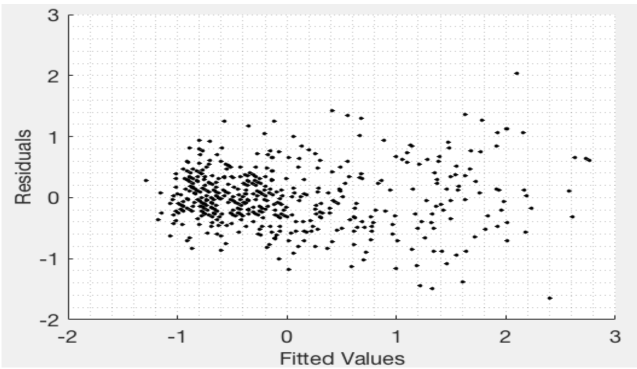

Figure 7: Fitted values vs Residuals

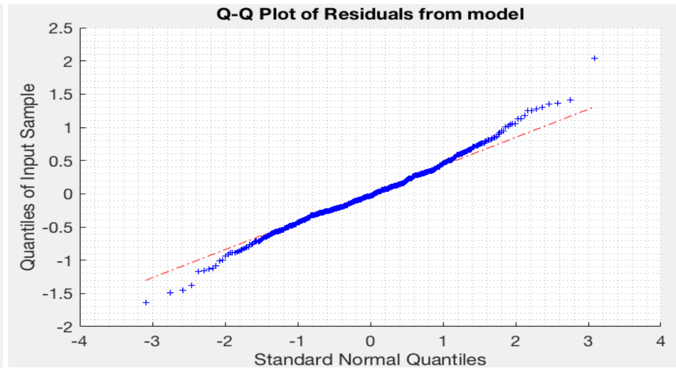

Figure 8: Q-Q plot of residuals

The residuals have no particular relationship with the fitted values, which shows that there is no systematic error throughout the model. Furthermore, it is assumed that the error term in a linear regression model has a distribution of $N(0, 1)$.

Interpreting regression coefficients

The regression coefficients for each of the predictor terms shows how much change there is in fuel efficiency when there is one unit of change in the predictor. Furthermore, the 'intercept' value is the value of fuel efficiency when all the predictor terms are set to zero. To make these coefficients more meaningful, all the values could be arithmetically standardized[1] which would mean the coefficients could be directly compared. This has already been done for the model in Question 2, and so is not necessary to do again.

4 Question 4

For the bootstrapping part of the assignment, the instructions (as given in the coursework handout) were followed. A new response variable was calculated using bootstrapped residuals from the model in Question 2 the same model was used to predict this new data. The aforementioned model is stated below:

$$l100 \sim 1 + mass + t100 + type + mass : t100 + mass : type + t100 : type$$

This process was repeated 10 times, and each time an array of fuel efficiency estimates (predicted using the linear model) for each acceleration time value, was calculated. Then, the mean value and the 0.025

[1] To standardize a term x_i, one can do

$$z_i = \frac{x_i - \mu_x}{\sigma_x}$$

and 0.975 quantiles for each acceleration time were calculated, and are plotted on the following page (Figures 9 and 10).

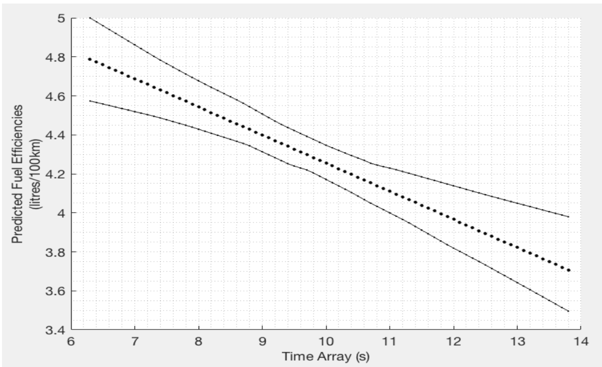

Figure 9: Predicted fuel efficiency by acceleration time, and bootstrapped 95% confidence band.

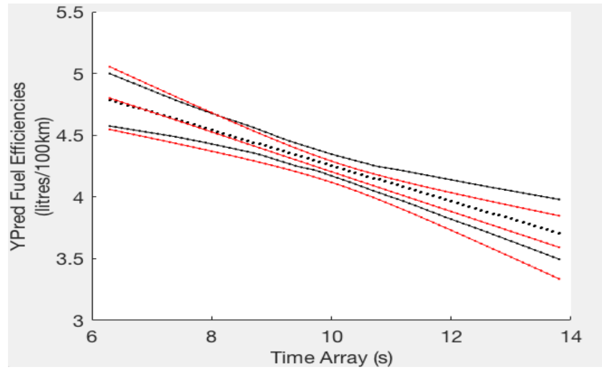

Figure 10: Model using 'predict' function (red) plotted alongside bootstrapped model.

It was found that, as the number of repeats for the process increased, the bootstrapped data would be more similar to the model obtained by the 'predict' function on MATLAB.

5 Question 5

Fitting the model from Question 2 to the test data was yielded a range of results. These are given below"

Training Data			Test Data		
R Squared	MSE	AIC	R Squared	MSE	AIC
0.764	0.2391	710.4619	0.7638	0.4061	975.3545

Therefore, the test data did not perform as well as the training data did; however, the values are not far off each other. Furthermore, the mean squared error value for the test data is larger than the same value for the training data, which suggests that there is little to no overfitting. Based on this evidence, it may be worth considering a slightly more complex model with more interaction terms.

By creating a linear model using this data, with quadratic terms and interactions (and no refining), we are able to achieve an R squared value of 0.774. However, the MSE is still larger than the simplified training data value (test data MSE is 0.3987) and the AIC value is 981.855, which is also higher than the corresponding value for the training data

6 Appendix - Code

```

%% ME3 STATS COURSEWORK
%AROHAN SUBRAMONIA
rng(01054062);

clear all
%% Reading the data from the csv file
AS10415 = readtable('as10415.csv');
DATA=table2cell(AS10415);
num=cell2mat(DATA(:,1:4));
num(:,5) = string(DATA(:,5)) == 'petrol'; %Fuel type logical (0 if Diesel, 1 if Petrol)
num(:,6) =  string(DATA(:,6)) == 'white';  %Colour logical (1 if white, 0 if other)

% Splitting the data up into four sections
l100=num(:,1);
mass=num(:,2);
t100=num(:,3);
disp=num(:,4);

%% Question 1
%% Exploratory Data Analysis
A_mean(1:4) = mean(num(:,1:4)); %Arithmetic mean
G_mean(1:4) = geomean(num(:,1:4)); %Geometric mean
median(1:4) = median(num(:,1:4)); %Median
T_mean(1:4) = mean(num(51:450,1:4)); %10% trimmed mean
A_std(1:4)  = std(num(:,1:4)); %Arithmetic standard deviation
top(1:4)    = sum(log(num(:,1:4)/G_mean).^2); %Geometric standard deviation
G_std(1:4)  = exp((top/499).^(0.5));

%% Histogram of Fuel Efficiency
histogram(num(:,1));
grid ON
fs=20; %FontSize
%title('Histogram of Fuel Efficiency per 100km','FontSize',18);
set(gca,'FontSize',fs)
xlabel('Fuel Efficiency (litres/100km)','FontSize',fs);
ylabel('Frequency','FontSize',fs);

%% Scatterplots
subplot(3,2,1), scatter(l100,mass,'.','k'), xlabel({'Fuel Efficiency';'(litres/100km)'},'FontSize',fs), ylabel('Vehicle Mass (kg)','FontSize',fs), grid MINOR
```

```
subplot(3,2,2), scatter(l100,t100,'.','k'), xlabel({'Fuel Efficiency';'(litres/100km)'},'FontSize',fs), ylabel({'Time taken to reach';'100km/h (s)'},'FontSize',fs), grid MINOR
subplot(3,2,3), scatter(l100,disp,'.','k'), xlabel({'Fuel Efficiency';'(litres/100km)'},'FontSize',fs), ylabel('Engine Size (litres)','FontSize',fs), grid MINOR
subplot(3,2,4), scatter(mass,t100,'.','k'), xlabel('Vehicle Mass (kg)','FontSize',fs), ylabel({'Time taken to reach';' 100km/h (s)'},'FontSize',fs), grid MINOR
subplot(3,2,5), scatter(mass,disp,'.','k'), xlabel('Vehicle Mass (kg)','FontSize',fs), ylabel('Engine Size (litres)','FontSize',fs), grid MINOR
subplot(3,2,6), scatter(t100,disp,'.','k'), xlabel({'Time taken to reach';' 100km/h (s)'},'FontSize',fs), ylabel('Engine Size (litres)','FontSize',fs), grid MINOR

%% Boxplot drawing routine
fueltype=[{['Petrol'] ['Diesel']}; {['White'] ['Other']}]; %Box labels

XLabel=[{'Fuel Type'} {'Fuel Color'}];
YLabel=[{['Fuel Efficiency'];[' (litres/100km)']};{'Vehicle Mass';'(kg)'};{'Time taken to';'reach 100km/h (s)'};{'Engine size';'(litres)'}];
yl=[1,1,3,3,5,5,7,7];
var=[1,1,2,2,3,3,4,4];
fs=18;
b=[1,2,1,2,1,2,1,2];
for a=1:8
    subplot(4,2,a), boxplot(num(:,var(a)),num(:,b(a)+4),'labels',fueltype(b(a),:));
    xlabel(XLabel(b(a)),'FontSize',fs);
    ylabel(YLabel(yl(a):yl(a)+1),'FontSize',fs);
    set(gca,'FontSize',fs)
end

%% Question 2
%% Standardizing and Organising data
Nl100=zscore(num(:,1));
Nmass=zscore(num(:,2));
Nt100=zscore(num(:,3));
Ndisp=zscore(num(:,4));
type = num(:,5);
colour=num(:,6);
%Creating a table of standardized variables
Nvarnames(1:6)=[{'Nl100'} {'Nmass'} {'Nt100'} {'Ndisp'} {'type'} {'colour'}];
Nnum=table(Nl100,Nmass,Nt100,Ndisp,num(:,5),num(:,6),'VariableNames',
```

```
    Nvarnames);

%% Linear Models - single variables without interaction

%Linear model of l100 vs mass
LMmass1=fitlm(mass,l100,'linear');
plot(LMmass1);
set(gca,'FontSize',fs)
ylabel('Fuel Efficiency (litres/100km)','FontSize',fs);
xlabel('Vehicle Mass (kg)','FontSize',fs);
title(' ')

%Quadratic model of l100 vs mass
LMmass2=fitlm(mass,l100,'purequadratic');
subplot(), plot(LMmass2);
set(gca,'FontSize',fs)
ylabel('Fuel Efficiency (litres/100km)','FontSize',fs);
xlabel('Vehicle Mass (kg)','FontSize',fs);

%Linear model of l100 vs t100
LMtime1=fitlm(t100,l100,'linear');
plot(LMtime1);
set(gca,'FontSize',fs)
ylabel('Fuel Efficiency (litres/100km)','FontSize',fs);
xlabel('Time taken to reach 100km/h (s)','FontSize',fs);

%Quadratic model of l100 vs t100
LMtime2=fitlm(t100,l100,'purequadratic');
plot(LMtime2);
set(gca,'FontSize',fs)
ylabel('Fuel Efficiency (litres/100km)','FontSize',fs);
xlabel('Time taken to reach 100km/h (s)','FontSize',fs);

%Linear model of l100 vs disp
LMdisp1=fitlm(disp,l100,'linear');
plot(LMdisp1);
set(gca,'FontSize',fs)
ylabel('Fuel Efficiency (litres/100km)','FontSize',fs);
xlabel('Engine Size (litres)','FontSize',fs);

%Quadratic model of l100 vs disp
LMdisp2=fitlm(disp,l100,'purequadratic');
plot(LMdisp2);
set(gca,'FontSize',fs)
ylabel('Fuel Efficiency (litres/100km)','FontSize',fs);
xlabel('Engine Size (litres)','FontSize',fs);

```

```
%% Parameters for single variable models without interaction (not
    standardized)
%Parameters for l100 vs mass - linear
Mass_linear=fitlm(Nmass,Nl100,'linear');
RSm1=Mass_linear.Rsquared.Ordinary;
MSEm1=Mass_linear.MSE;
AICm1=Mass_linear.ModelCriterion.AIC;

%Parameters for l100 vs mass - quadratic
Mass_quad=fitlm(Nmass,Nl100,'purequadratic');
RSm2=Mass_quad.Rsquared.Ordinary;
MSEm2=Mass_quad.MSE;
AICm2=Mass_quad.ModelCriterion.AIC;

%Parameters for l100 vs time - linear
Time_linear=fitlm(Nt100,Nl100,'linear');
RSt1=Time_linear.Rsquared.Ordinary;
MSEt1=Time_linear.MSE;
AICt1=Time_linear.ModelCriterion.AIC;

%Parameters for l100 vs time - quadratic
Time_quad=fitlm(Nt100,Nl100,'purequadratic');
RSt2=Time_quad.Rsquared.Ordinary;
MSEt2=Time_quad.MSE;
AICt2=Time_quad.ModelCriterion.AIC;

%Parameters for l100 vs disp - linear
Disp_linear=fitlm(Ndisp,Nl100,'linear');
RSd1=Disp_linear.Rsquared.Ordinary;
MSEd1=Disp_linear.MSE;
AICd1=Disp_linear.ModelCriterion.AIC;

%Parameters for l100 vs disp - quadratic
Disp_quad=fitlm(Ndisp,Nl100,'purequadratic');
RSd2=Disp_quad.Rsquared.Ordinary;
MSEd2=Disp_quad.MSE;
AICd2=Disp_quad.ModelCriterion.AIC;

RES_RS=[RSm1;RSm2;RSt1;RSt2;RSd1;RSd2];
RES_MSE=[MSEm1;MSEm2;MSEt1;MSEt2;MSEd1;MSEd2];
RES_AIC=[AICm1;AICm2;AICt1;AICt2;AICd1;AICd2];

%Final table of single variable parameters
modelname={'mass (linear)','mass (quad)','time (linear)','time (quad)',
    'engine size (linear)','engine size (quad)'}';
estnames={'Type' 'Rsquared' 'MSE' 'AIC'};
statistics=table(modelname,RES_RS,RES_MSE,RES_AIC,'VariableNames',
```

```
    estnames);

%% Linear Models - multivariable, some with interactions

%Linear multivariate model, no interactions
Linear_Model = fitlm(Nnum,'linear','PredictorVars',{'Nmass','Nt100','
    Ndisp','type','colour'},'ResponseVar','Nl100');
lintab=table({'Linear Multivariate'},Linear_Model.Rsquared.Ordinary,
    Linear_Model.MSE,Linear_Model.ModelCriterion.AIC,'VariableNames',
    estnames);

%Quadratic multivariate model, no interactions
Quad_Model = fitlm(Nnum,'purequadratic','PredictorVars',{'Nmass','Nt100
    ','Ndisp','type','colour'},'ResponseVar','Nl100');
quadtab=table({'Pure Quadratic Multivariate'},Quad_Model.Rsquared.
    Ordinary,Quad_Model.MSE,Quad_Model.ModelCriterion.AIC,'VariableNames
    ',estnames);

%Linear multivariate model, with interactions
Inter_Model1 = fitlm(Nnum,'interactions','PredictorVars',{'Nmass','
    Nt100','Ndisp','type','colour'},'ResponseVar','Nl100');

%Quadratic multivariate model, with interactions
Inter_Model2 = fitlm(Nnum,'quadratic','PredictorVars',{'Nmass','Nt100',
    'Ndisp','type','colour'},'ResponseVar','Nl100');

inter1tab=table({'Linear Interactions'},Inter_Model1.Rsquared.Ordinary,
    Inter_Model1.MSE,Inter_Model1.ModelCriterion.AIC,'VariableNames',
    estnames);
inter2tab=table({'Quadratic Interactions'},Inter_Model2.Rsquared.
    Ordinary,Inter_Model2.MSE,Inter_Model2.ModelCriterion.AIC,'
    VariableNames',estnames);

%Final table of all models
statistics(7:10,1:4)=[lintab;quadtab;inter1tab;inter2tab];

%% Refining the best performing (quadratic) model

LM2=Inter_Model2;
%Removing the least significant coefficients
for a=1:(0.5*length(LM2.Coefficients.Estimate))
    [X,I]=min(abs(LM2.Coefficients.Estimate));
    if I ~= 1
        LM2=removeTerms(LM2,char(LM2.CoefficientNames(I)));
    end
end

```

```
for a=1:10
    LM2=step(LM2);
end

%% Adding terms in to finalise model
mdl_eqn2 = ('Nl100~1+Nmass+Nt100+type+Nmass:Nt100+Nmass:type+Nt100:type');
LM2=fitlm(Nnum,mdl_eqn2);

%% Question 3
%% Residual and Fitted Plot
res=LM2.Residuals.Raw;
fit=LM2.Fitted;

scatter(fit,res,10,'k','MarkerFaceColor','k','MarkerEdgeColor','k');
grid MINOR;
set(gca,'FontSize',fs);
xlabel('Fitted Values','FontSize',fs);
ylabel('Residuals','FontSize',fs);

%% QQ Plot of residuals
qqplot(res);
grid MINOR;
title('Q-Q Plot of Residuals from model','FontSize',fs);
set(gca,'FontSize',fs);

%% Coefficient Plot
coeff=LM2.Coefficients.Estimate;

plot(coeff,'k.-','MarkerSize',15);
grid MINOR;
set(gca,'FontSize',fs);
xlabel('Coefficients index','FontSize',fs);
ylabel('Coefficients values','FontSize',fs);

%% Question 4
%% Bootstrapping
mint=min(t100);
maxt=max(t100);
time=[mint:0.1:maxt];
res1=LM2.Residuals.Raw;
l100hat=LM2.Fitted;
res=l100-l100hat;

%Using lMstar, predict
for j=1:10
    res_star = randsample(res,length(res),'true');
```

```
    l100star=LM2.Fitted+res_star;

    % Same model as Q2 fitted to Nl100star
    mdl_eqnstar = ('l100star~1+mass+t100+type+mass:t100+mass:type+t100:type');
    numstar=table(l100star,mass,t100,disp,num(:,5),num(:,6),'VariableNames',{'l100star' 'mass' 't100' 'disp' 'type' 'colour'});
    LM2star=fitlm(numstar,mdl_eqnstar);

    for i=1:length(time)
        l100starhat(i)=LM2star.Coefficients.Estimate(1)+...
            LM2star.Coefficients.Estimate(2).*mean(mass)+...
            LM2star.Coefficients.Estimate(3).*time(i)+...
            LM2star.Coefficients.Estimate(4).*min(type)+...
            LM2star.Coefficients.Estimate(5).*(mean(mass).*time(i))+...
            LM2star.Coefficients.Estimate(6).*(mean(mass).*min(type))+...
            LM2star.Coefficients.Estimate(5).*(time(i).*min(type));
    end

    l100stardata(:,j)=l100starhat';
end

l100starmean=mean(l100stardata');
lower = quantile(l100stardata,0.025,2);
upper = quantile(l100stardata,0.975,2);

scatter(time,l100starmean,5,'MarkerFaceColor','k','MarkerEdgeColor','k');
hold;
plot(time,lower,'k.-',time,upper,'k.-');
grid MINOR
set(gca,'FontSize',fs)
xlabel('Time Array (s)','FontSize',fs);
ylabel({'Predicted Fuel Efficiencies';'(litres/100km)'},'FontSize',fs);

%% Using a separate model to predict confidence intervals

xnew=zeros(length(time),5);
xnew(:,1)=mean(mass);
xnew(:,2)=time;
xnew(:,3)=mean(disp);

[ypred,yci]=predict(LM2star,xnew);

plot(time,ypred,'r.-',time,yci,'r.-');
```

```
grid MINOR
set(gca,'FontSize',fs)
xlabel('Time Array (s)','FontSize',fs);
ylabel({'YPred Fuel Efficiencies';'(litres/100km)'},'FontSize',fs);

%% Question 5
%% Reading the data from the csv file
testdata = readtable('testdata.csv');
TDATA=table2cell(AS10415);
tnum=cell2mat(TDATA(:,1:4));
tnum(:,5) = string(TDATA(:,5)) == 'petrol'; %Fuel type logical (0 if Diesel, 1 if Petrol)
tnum(:,6) =  string(TDATA(:,6)) == 'white';  %Colour logical (1 if white, 0 if other)

% Splitting the data up into six sections
Tl100=tnum(:,1);
Tmass=tnum(:,2);
Tt100=tnum(:,3);
Tdisp=tnum(:,4);
Ttype=tnum(:,5);
Tcolour=tnum(:,6);

%% Using Question 2 model on Test data
Tvarnames(1:6)=[{'Tl100'} {'Tmass'} {'Tt100'} {'Tdisp'} {'Ttype'} {'Tcolour'}];
Tnum=table(Tl100,Tmass,Tt100,Tdisp,Ttype,Tcolour,'VariableNames',Tvarnames);

mdl_eqnT = ('Tl100~1+Tmass+Tt100+Ttype+Tmass:Tt100+Tmass:Ttype+Tt100:Ttype');
LMT=fitlm(Tnum,mdl_eqnT);

%Quadratic Interactions model on test data
Test_Inter_Quad_Model=fitlm(Tnum,'quadratic','PredictorVars',{'Tmass' 'Ttype' 'Tt100' 'Tdisp' 'Ttype' 'Tcolour'},'ResponseVar','Tl100');
```